Electronic Supplementary Material (ESI) for Materials Horizons.

Combining Electronic and Structural Features in Machine Learning Models to Predict Organic Solar Cells Properties

Daniele Padula,* Jack D. Simpson, Alessandro Troisi*

Department of Chemistry, University of Liverpool, Liverpool L69 7ZD, U.K.

E-mail: DP dpadula@liverpool.ac.uk, AT atroisi@liverpool.ac.uk

Supporting Information

Summary

Data Gathering

The database includes 249 Donor-Acceptor couples. Donors are small organic molecules (see Fig. S3 for some examples and the list of donors at the end of this document), while Acceptors are fullerenes, namely C_{60}, $PC_{61}BM$, and $PC_{71}BM$. Optimised structures, literature sources, experimental and computed properties are available as Supporting Information. Searches of the literature were conducted using Scopus. "Organic Solar Cells" was used as the keyword, refining the search to only include review articles between the years 2013 and 2017, obtaining 656 reviews. Any review articles with “polymer” in the title were ignored. For each result, the review was searched for reporting of experimental values of PCE, J_{sc}, V_{oc} and FF. For each paper that reported experimental values, the impact factor of the review was found, and it was only considered if it had an impact factor greater than 2.5. While this selection criterion can be debatable, it ensures an unbiased and reproducible data gathering process. Moreover, the value chosen for the impact factor threshold allows to include contributions from very technical journals (*e.g. The Journal of Physical Chemistry, Physical Chemistry Chemical Physics*), thus we deem it acceptable. Finally it also allows to overcome a possible sampling problem that might arise from the inclusion of too many data points at low photovoltaic efficiencies. Overall, 180 reviews were sourced leaving 476 reviews. For each device reported in the review, the original paper was searched, and all donors reported in that paper were recorded. If a range of values were reported, the average would be taken as well as a note of any experimental conditions or ratios reported. If any duplicates were found, they were removed from the database. Moreover, donors were only recorded if their corresponding acceptor was a fullerene and the composition of the active layer of the solar cells was a bulk heterojunction or bilayer heterojunction. As side chains in semiconductor devices are known to have little effect on electronic properties, each molecule containing long alkyl chains had their side chains were cut using the same methodology reported previously.

Data set download link: https://bitbucket.org/dpadula85/organic-semiconducting-donors/src

Computational Details

Optimised geometries and electronic properties were computed *in vacuo* at DFT/B3LYP/6-31G* level of theory using the Gaussian 16 software.[1] Reorganisation energies were obtained computing energy differences among single point calculations of the species involved in the electron transfer process (also known as “4-point strategy”).[2] Geometry optimisations were repeated including the PCM solvation model through its Integral Equation Formalism for toluene and chloroform. The effect of the solvent on frontier orbital energies and geometries is negligible and summarised in Fig. S11. Molecular fingerprints were obtained by converting optimised geometries to a 2-D representation through the OpenBabel software,[3] followed by generation of molecular fingerprints as implemented in the RDKit package.[4]

The Scharber open circuit voltage V_{OC}^{Sch} was calculated by using the extracted values of HOMO and LUMO for the donor and acceptor respectively. To calculate the Scharber short circuit current J_{SC}^{Sch}, the energy difference between the HOMO and LUMO of the donor

molecule was calculated. From this, the AM1.5G reference spectrum was integrated with respect to wavelength between zero and the wavelength of a photon at the energy of the donor's gap. The Scharber PCE η^{Sch} was then calculated according to Eq. 2 in the main text, using P_{in}=1000 W m^{-2}.

Predictions of PCE have been computed by adopting Scharber's model and various Machine Learning algorithms by using as input (i) DFT electronic features (namely E_D^{HOMO}, E_D^{LUMO}, E_A^{LUMO}, and the reorganisation energy λ) (ii) molecular fingerprints.

For machine learning models, the best hyperparameters were identified minimising the average cross-validation RMSE, adopting a Differential Evolution optimisation algorithm[5] in combination with a LOO cross-validation procedure. This means that, for each set of hyperparameters, each of the points of the data set is predicted by training a model on the remaining N-1 points, thus the average RMSE results from training N different models (N size of the data set). For the stochastic optimisation, we used a population size of 100, with a recombination rate of 0.7 and a mutation constant randomly picked between 0.5 and 1. The bounds for hyperparameters have been chosen as 0 < γ_1, γ_2 < 10 and 0 < α < 2. The optimal combinations of hyperparameters are reported in the Supporting Information. KRR predictions are the LOO predictions computed with the optimal hyperparameters.

Kernel Ridge Regression

KRR is a modified version of regularised least squares (also known as Ridge regression) that allows introducing non-linearity thanks to the so-called kernel trick.[6-7] KRR is equivalent to Gaussian Processes Regression (GPR) if one assumes the error on the predictions to be negligible. Given a training set of *N* examples $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, we denote as $\mathbf{x}_i$ the column vector containing *d* inputs for the *i*-th example, and as y_i the output (*i.e.* the value of the target experimental property). We gather all the vectors $\mathbf{x}_i$ in a matrix X with *N* rows and *d* columns (*i.e.* each row of X contains an example), and outputs y_i in a column vector $\mathbf{y}$. The least squares solution to the linear fit $X\mathbf{w} = \mathbf{y}$ can be written as

$$\mathbf{w} = (X^T X)^{-1} X^T \mathbf{y} \tag{1}$$

To avoid overfitting, it is convenient to add a regularisation term to the least squares objective function to be minimised, to penalise high-valued solutions. The solution to the regularised problem reads

$$\mathbf{w} = (X^T X + \alpha I)^{-1} X^T \mathbf{y} = X^T (X X^T + \alpha I)^{-1} \mathbf{y} \tag{2}$$

where α is the regularisation hyperparameter, to be chosen empirically or by cross-validation. To obtain the predictions $\mathbf{y}'$ for a new point $\mathbf{x}'$, we should compute

$$\mathbf{y}' = \mathbf{w}^T \mathbf{x}'^T = \mathbf{y}^T (X X^T + \alpha I)^{-1} X \mathbf{x}'^T = \mathbf{y}^T (K + \alpha I)^{-1} \kappa' \tag{3}$$

where we defined the matrix $K = XX^T$, with elements $K_{ij} = \mathbf{x}_i^T \mathbf{x}_j$, and the matrix $\kappa' = X\mathbf{x}'^T$, with elements $\kappa'_i = \mathbf{x}_i^T \mathbf{x}'$. To introduce non-linearity, *i.e.* replace each element $\mathbf{x}_i$ with a vector function $\boldsymbol{\phi}_i = f(\mathbf{x}_i)$ and do the fitting with the $\boldsymbol{\phi}$ vectors, it is possible to adopt the kernel trick by replacing K and κ' with the following

$$K_{ij} = f(\mathbf{x}_i, \mathbf{x}_j)$$
$$\kappa'_i = f(\mathbf{x}_i, \mathbf{x}') \quad (4)$$

where f is a function that maps two vectors to a scalar. The particular form of f also determines the non-linearity introduced in the procedure. A kernel matrix can be interpreted as matrix of distances between couples of examples in an input space.

Prediction of Photovoltaic Properties for Scharber's Model

Given the two distance measures expressed in Eqs. 2-3 in the main text, we can define the kernel function f as including a linear combination of the two distances, weighted by two hyperparameters γ_1 and γ_2 to be chosen through cross-validation. Thus, we can rewrite Eq. 4 as

$$K_{ij} = f(\mathbf{x}_i, \mathbf{x}_j) = e^{-(\gamma_1 D_{el}^2(\mathbf{x}_i, \mathbf{x}_j) + \gamma_2 D_{fp}^2(\mathbf{x}_i, \mathbf{x}_j))}$$
$$\kappa'_i = f(\mathbf{x}_i, \mathbf{x}') = e^{-(\gamma_1 D_{el}^2(\mathbf{x}_i, \mathbf{x}') + \gamma_2 D_{fp}^2(\mathbf{x}_i, \mathbf{x}'))} \quad (5)$$

This allows to use either electronic or structural information only, by setting $\gamma_2 = 0$ or $\gamma_1 = 0$ respectively, or to include both electronic and structural information in the model. Notice that if structural information are neglected by setting $\gamma_2 = 0$, Eq. 5 corresponds to adopting a Radial Basis Function kernel.

To compare with previous works where Scharber's model input were refined through kernel methods, we employed Eq. 5 setting $\gamma_1 = 0$ and thus only relying on structural information. We obtained computed photovoltaic parameters (V_{OC}^{cal}, J_{SC}^{cal}) to be used as input for Scharber's model. More in detail, for each point in the data set we define a training set as the remaining *N*-1 examples, for which we can define $\mathbf{y}$ as the experimental values of the target property for the training set. We then obtained the kernel matrices K and $\kappa(x)$ according to Eq. 5, and the prediction on the desired point according to Eq. 3. This procedure, known as Leave-One-Out (LOO), resulted in the training of *N* different models. The hyperparameters α and γ_2 (we set $\gamma_1 = 0$) were obtained by a stochastic minimisation of the average RMSE given by the *N* different models (see Computational Details). The efficiencies are finally obtained applying Scharber's model to the computed photovoltaic parameters V_{OC}^{cal}, J_{SC}^{cal}.

The results of this procedure are reported in Fig. S1. In comparison to Fig. 1 in the main text, we notice that this strategy results in a better agreement between computed properties and experimental ones, especially when using Morgan fingerprints. In comparison to the analogous *k*-NN predictions reported in Fig. 3, the results obtained through this procedure are similar: despite the fact that here we obtain better values to use as input for Scharber's model, especially for V_{OC}^{cal}, the difference is not big enough to result in a significant improvement of predicted efficiencies.

In comparison to previous works where similar kernel methods have been used adopting electronic parameters as input, we obtain comparable results, in terms of correlation metrics and RMSE (see Table S1). It is important to notice that the computational effort required for the calibration procedure previously reported is much higher, as one needs input DFT data. The predictions in Figs. 3 and S1 do not need any computational input, other than molecular fingerprints.

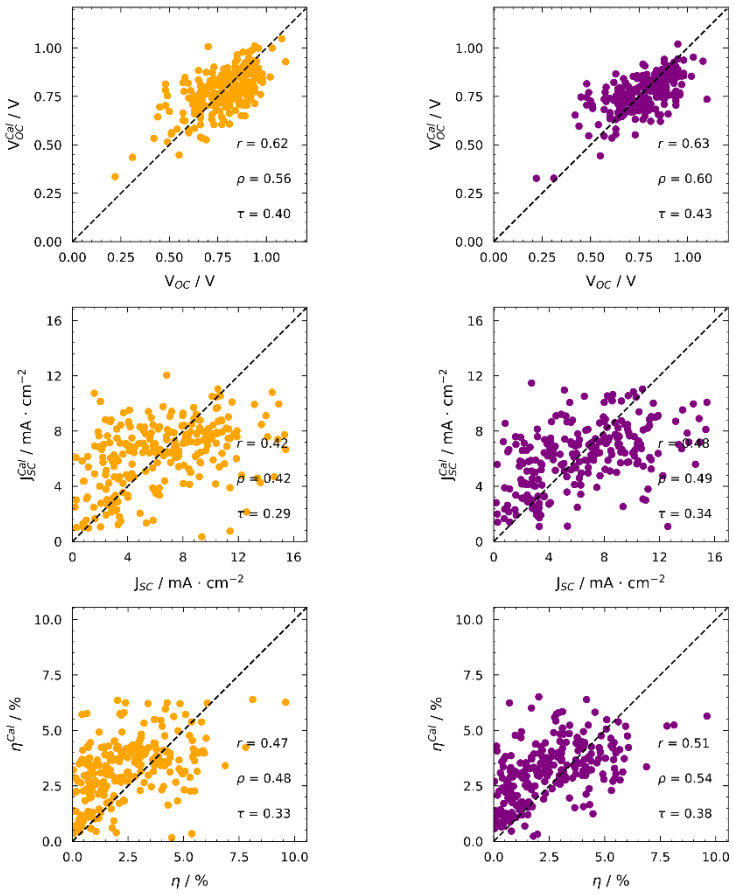


Fig. S1: KRR predictions of photovoltaic properties (V_{OC}^{cal}, J_{SC}^{cal}) from geometric similarity. Efficiencies have been computed by using the computed photovoltaic properties as input for Scharber's model. Left: Tanimoto similarity index from Daylight fingerprints. Right: Tanimoto similarity index from Morgan fingerprints.

Figures

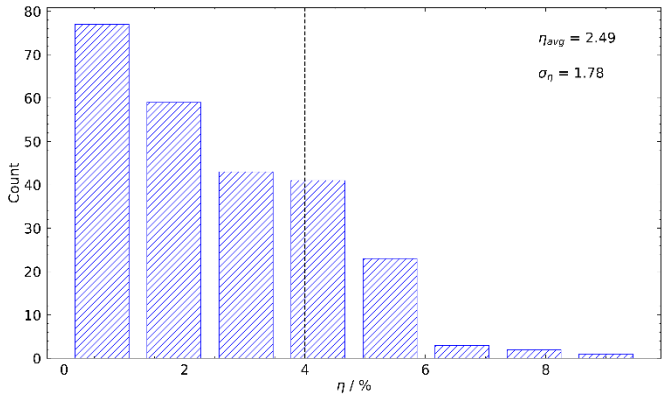


Fig. S2: Distribution of experimental efficiencies in the data set.

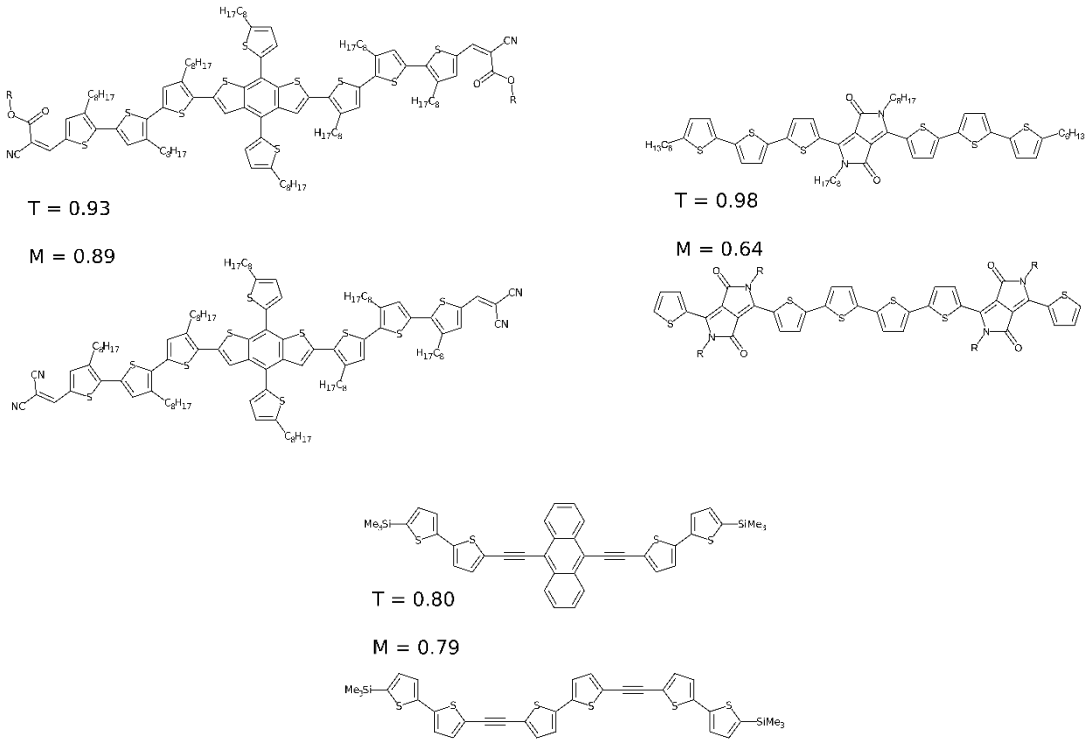


Fig. S3: examples of pairs of similar Donors in the database with their Tanimoto similarity index $T(\mathbf{x}_i^{fp}, \mathbf{x}_j^{fp})$ computed using Daylight (T) and Morgan (M) fingerprints. The reported examples highlight how the similarity index based on Morgan fingerprints is able to capture better structural similarity.

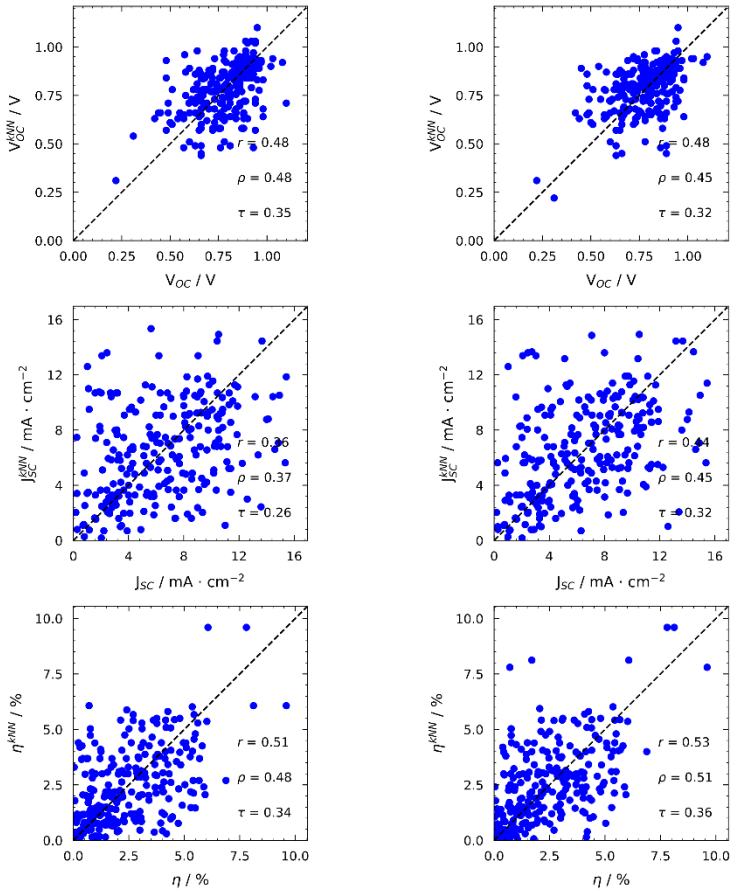


Fig. S4: *k*-NN regression predictions of photovoltaic parameters (*k* = 1). Left: Tanimoto similarity index from Daylight fingerprints. Right: Tanimoto similarity index from Morgan fingerprints.

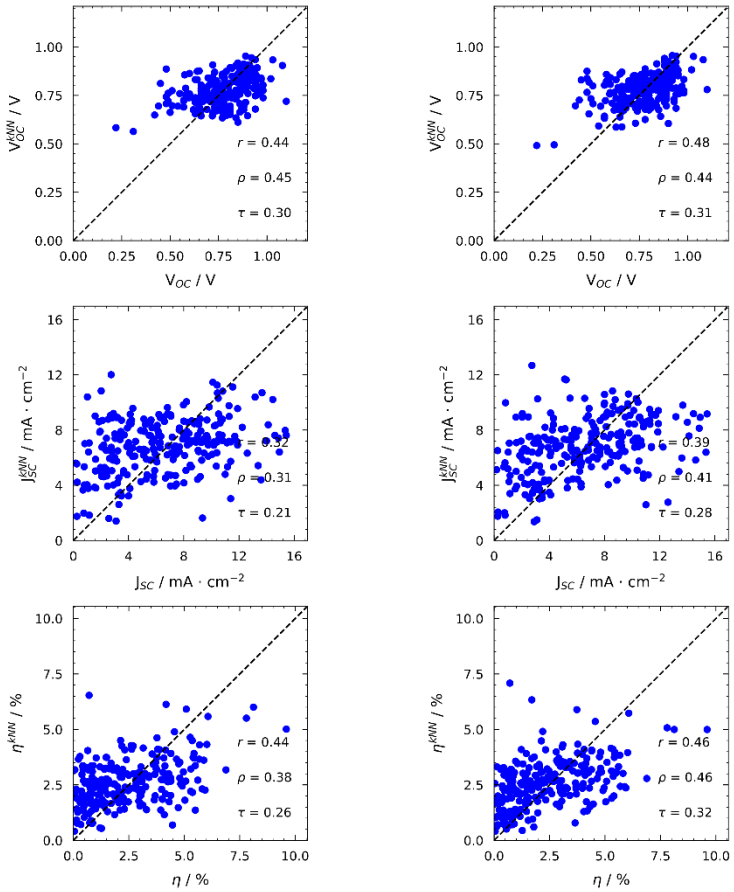


Fig. S5: *k*-NN regression predictions of photovoltaic parameters (*k* = 5). Left: Tanimoto similarity index from Daylight fingerprints. Right: Tanimoto similarity index from Morgan fingerprints.

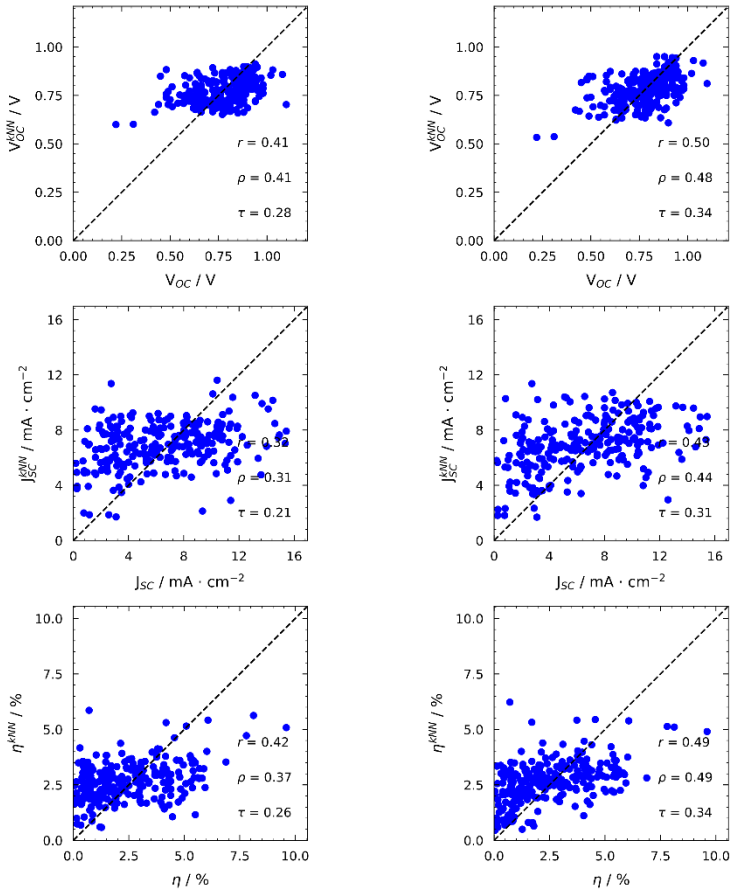


Fig. S6: *k*-NN regression predictions of photovoltaic parameters (*k* = 7). Left: Tanimoto similarity index from Daylight fingerprints. Right: Tanimoto similarity index from Morgan fingerprints.

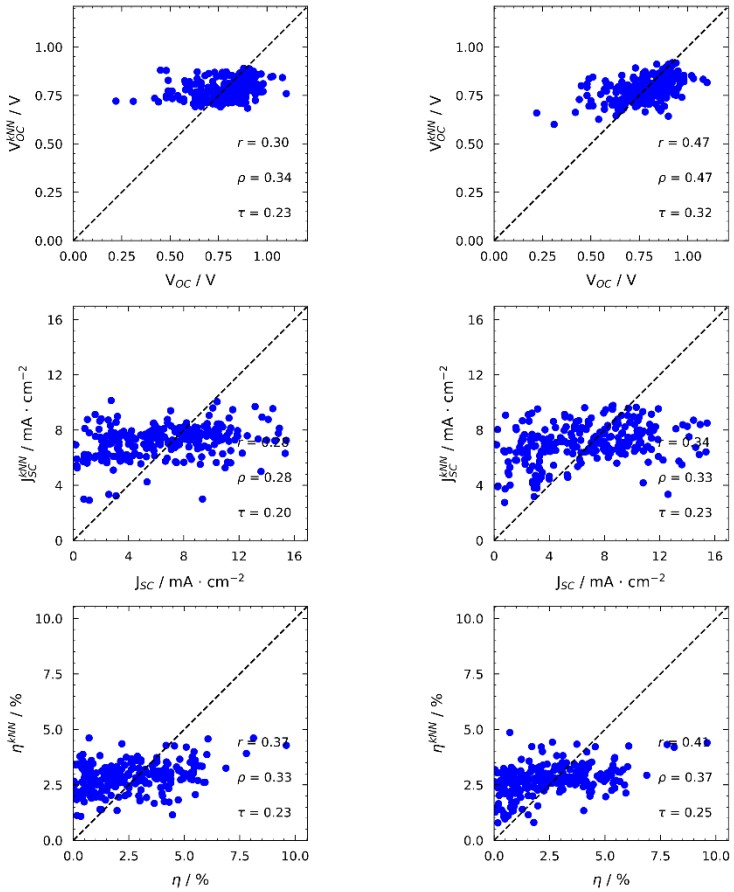


Fig. S7: *k*-NN regression predictions of photovoltaic parameters (*k* = 15). Left: Tanimoto similarity index from Daylight fingerprints. Right: Tanimoto similarity index from Morgan fingerprints.

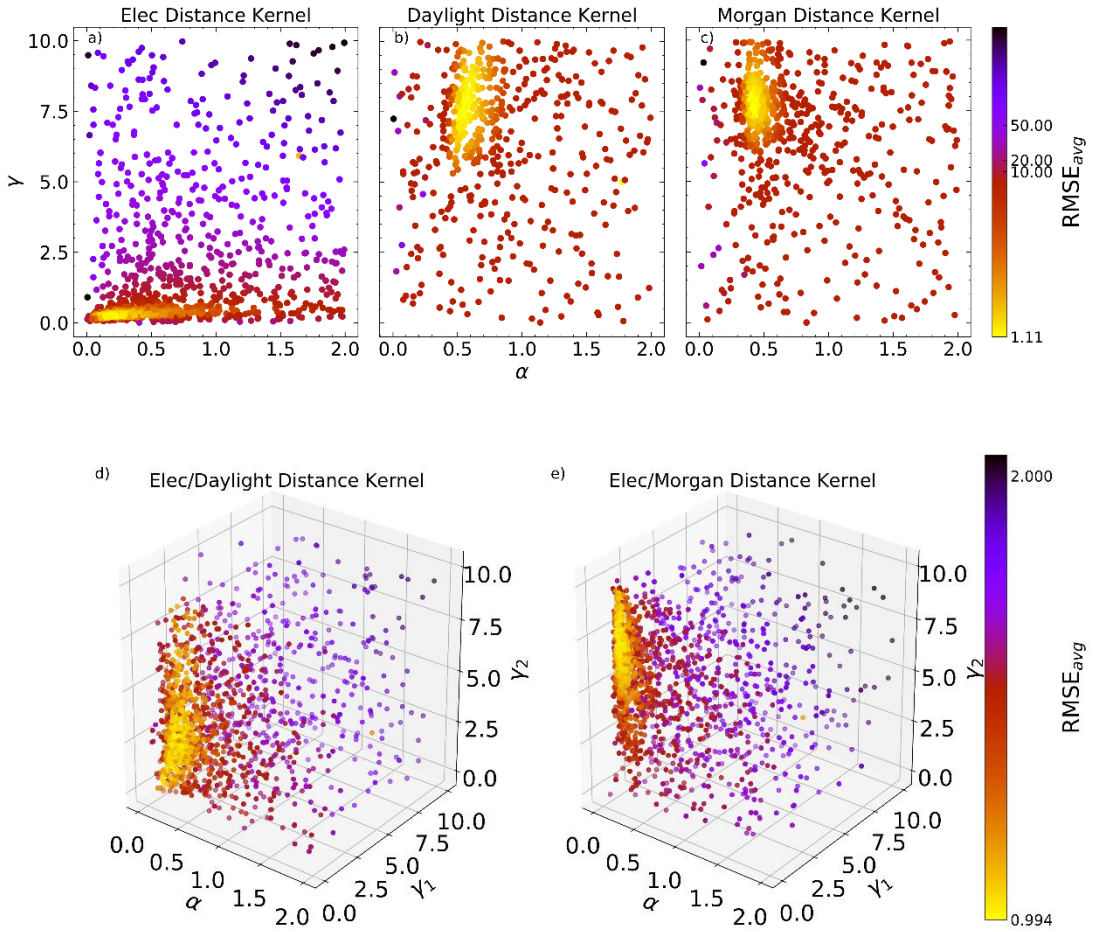


Fig. S8: average LOO RMSE as a function of hyperparameters for KRR models for photovoltaic efficiency prediction. Optimal hyperparameters have been identified through stochastic optimisation (Differential Evolution). Each point represents the average of 249 KRR models trained with a particular combination of hyperparameters.

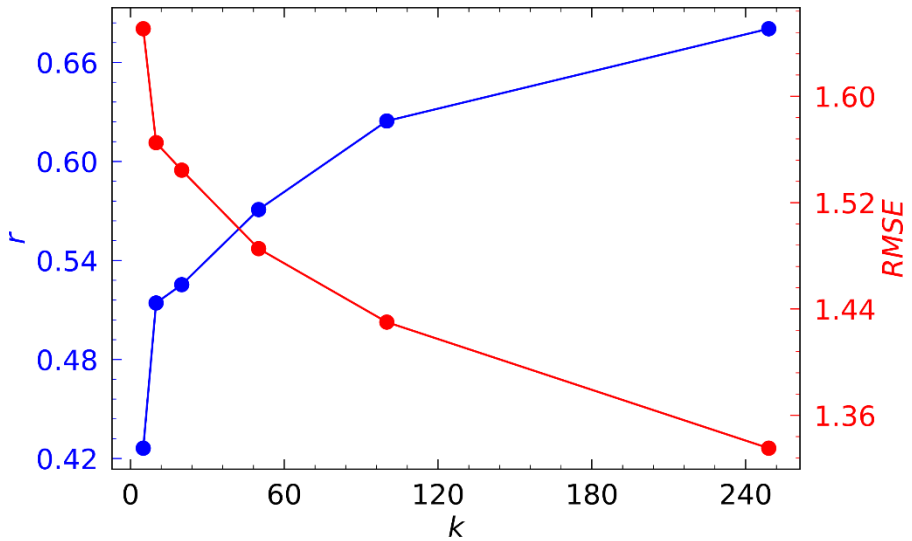


Fig. S9: effect of number of splits in *k*-fold cross validation on the performance of the KRR model based on Elec/Morgan Distance Kernel.

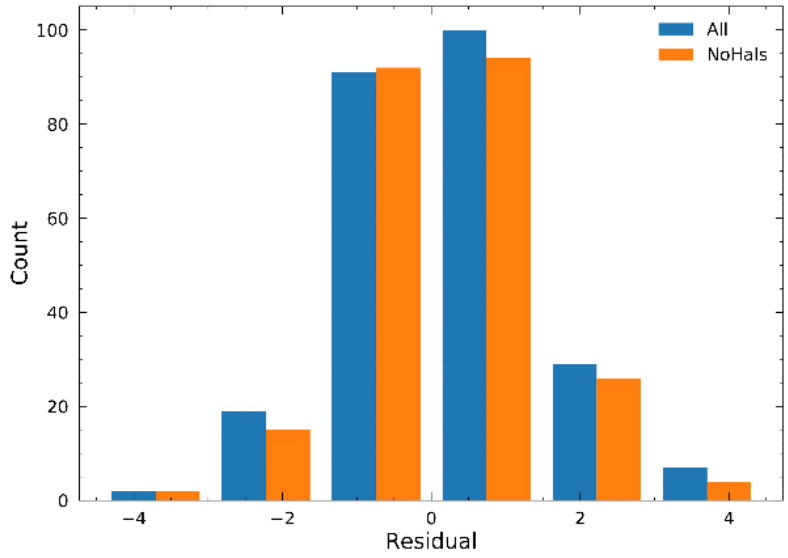


Fig. S10: Distribution of residual errors for the KRR model based on Elec/Morgan Distance Kernel trained on all dataset entries (blue), and excluding entries containing halogen atoms (orange). The two groups have been likely drawn from the same distribution (Kruskal-Wallis-p = 0.96).

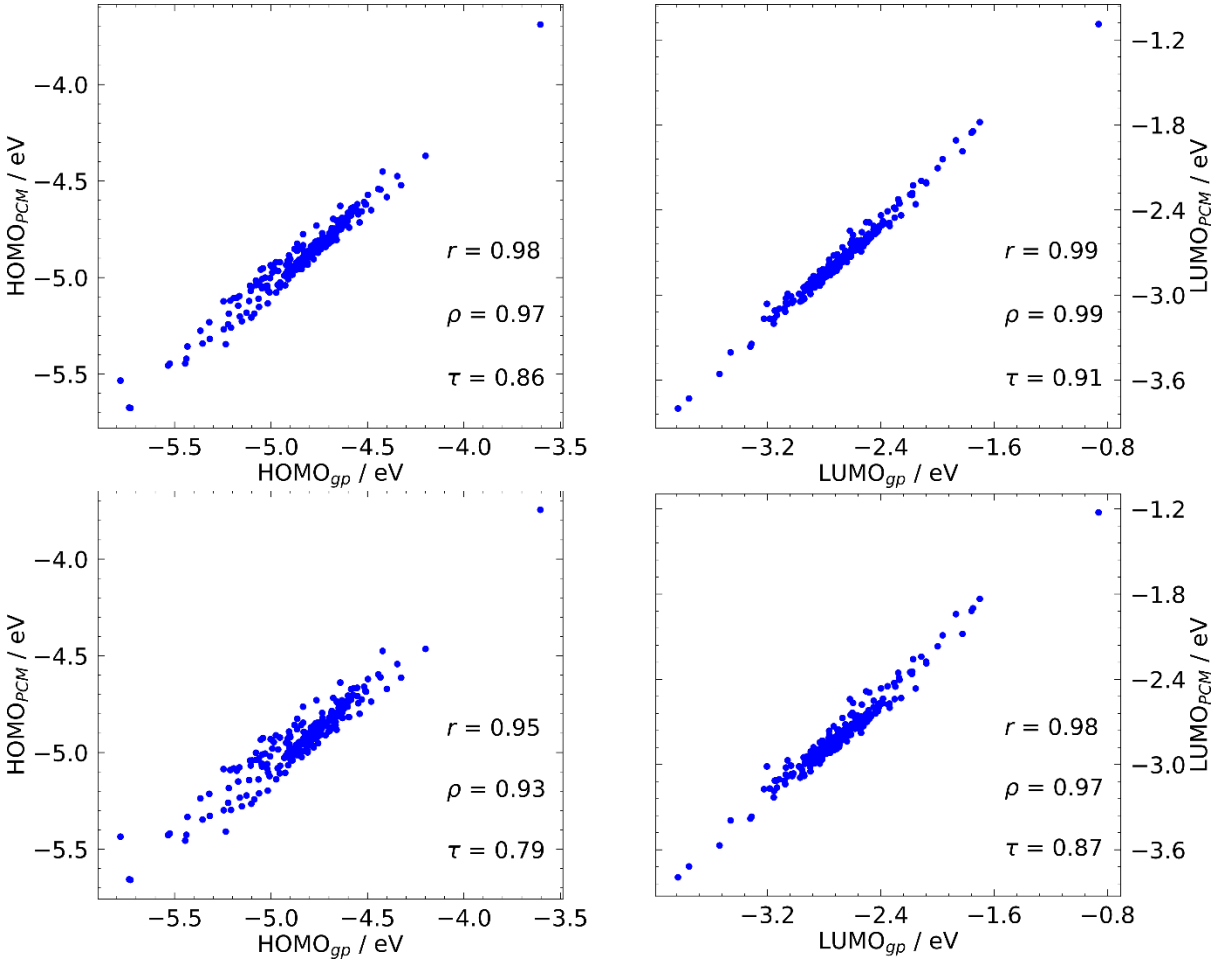


Fig. S11: effect of PCM solvation model in geometry optimisations on frontier orbitals energies for the donors in the data set (top: toluene, bottom: choloroform).

Tables

Table S1: Values of best hyperparameters and RMSE for the models used. Units for RMSE are consistent with the units of the targeted experimental property (V for V_{OC}, mA · cm^{-2} for J_{SC}, % for η). The best model for predictions of η is highlighted in bold.

Fig.	Model	Input	Target Property	α	γ_1	γ_2	RMSE
3	*k*-NN	Elec	V_{OC}	/	/	/	0.1272
3	*k*-NN	Elec	J_{SC}	/	/	/	3.5988
3	*k*-NN	Elec	η	/	/	/	1.7808
3	*k*-NN	Daylight	V_{OC}	/	/	/	0.1239
3	*k*-NN	Daylight	J_{SC}	/	/	/	3.5855
3	*k*-NN	Daylight	η	/	/	/	1.5834
3	*k*-NN	Morgan	V_{OC}	/	/	/	0.1207
3	*k*-NN	Morgan	J_{SC}	/	/	/	3.3571
3	*k*-NN	Morgan	η	/	/	/	1.6263
3	*k*-NN	Elec/Daylight	V_{OC}	/	0.6104	6.7108	0.1137
3	*k*-NN	Elec/Daylight	J_{SC}	/	0.7129	8.1892	3.0464
3	*k*-NN	Elec/Daylight	η	/	0.3637	4.4804	1.4585
3	*k*-NN	Elec/Morgan	V_{OC}	/	2.5127	7.6018	0.1037
3	*k*-NN	Elec/Morgan	J_{SC}	/	1.8503	7.1529	3.1246
3	*k*-NN	Elec/Morgan	η	/	2.7856	9.8470	1.4848
S1	KRR	Daylight	V_{OC}	0.0446	0.0000	0.5622	0.1092
S1	KRR	Daylight	J_{SC}	0.2666	0.0000	5.0595	3.4006
S1	Scharber	Daylight $V_{OC}^{cal}, J_{SC}^{cal}$	η	/	/	/	1.8359
S1	KRR	Morgan	V_{OC}	0.4216	0.0000	3.1366	0.1062
S1	KRR	Morgan	J_{SC}	0.6287	0.0000	6.3331	3.2238
S1	Scharber	Morgan $V_{OC}^{cal}, J_{SC}^{cal}$	η	/	/	/	1.7120
4	KRR	Daylight	η	0.5331	0.0000	7.5510	1.6383
4	KRR	Morgan	η	0.4169	0.0000	7.5865	1.4872
4	KRR	Elec	η	0.1954	0.2497	0.0000	1.5575
4	KRR	Elec/Daylight	η	0.3071	0.3500	2.9579	1.4515
4	**KRR**	**Elec/Morgan**	$\boldsymbol{\eta}$	**0.2144**	**0.1381**	**7.7656**	**1.3353**

The best model shows an RMSE≈1.33% (see Table S1). We compare this result with the RMSE one would obtain with predictions always equal to the experimental average, which is RMSE≈1.78%, a surprisingly low value due to the fact that the distribution of experimental efficiencies is strongly skewed towards low values (see Fig. S2).

Table S2: Values of correlation coefficients and RMSE for the models used. The best model for predictions of η is highlighted in bold.

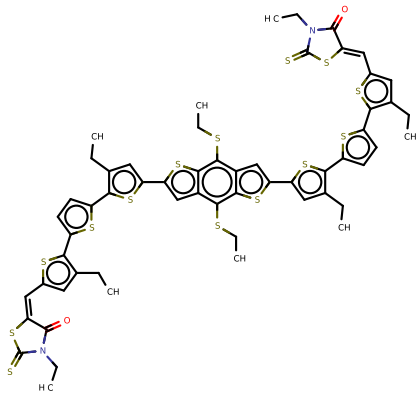
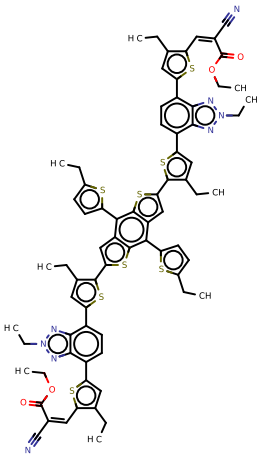
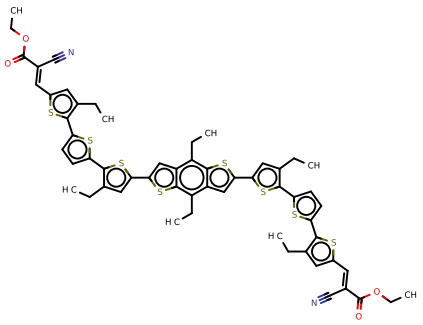
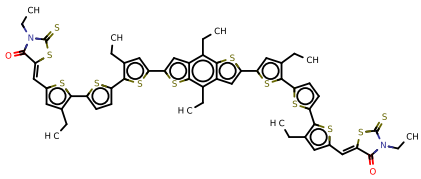
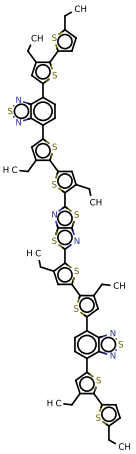
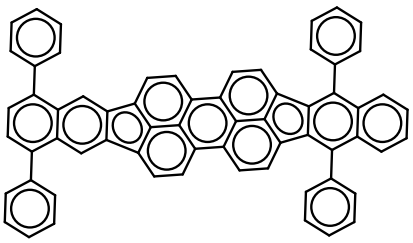
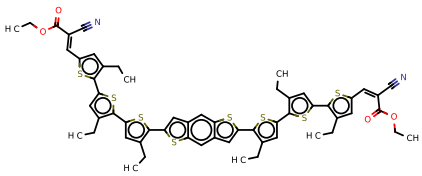
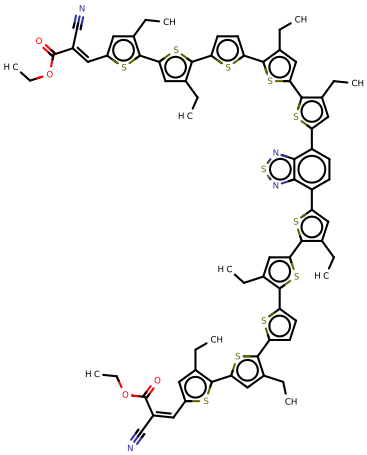
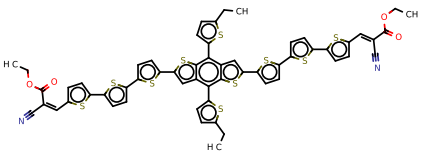
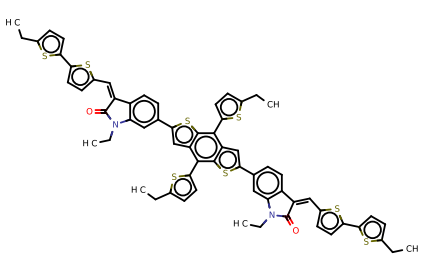
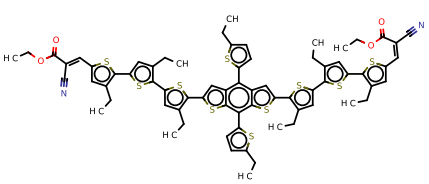
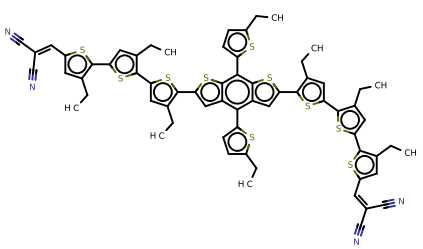
Fig.	Model	Input	Target Property	r	ρ	τ
3	*k*-NN	Elec	V_{OC}	0.43	0.43	0.31
3	*k*-NN	Elec	J_{SC}	0.37	0.37	0.25
3	*k*-NN	Elec	η	0.38	0.35	0.24
3	*k*-NN	Daylight	V_{OC}	0.48	0.47	0.33
3	*k*-NN	Daylight	J_{SC}	0.36	0.33	0.23
3	*k*-NN	Daylight	η	0.51	0.43	0.30
3	*k*-NN	Morgan	V_{OC}	0.50	0.47	0.33
3	*k*-NN	Morgan	J_{SC}	0.45	0.47	0.33
3	*k*-NN	Morgan	η	0.49	0.51	0.35
3	*k*-NN	Elec/Daylight	V_{OC}	0.57	0.56	0.40
3	*k*-NN	Elec/Daylight	J_{SC}	0.57	0.59	0.42
3	*k*-NN	Elec/Daylight	η	0.61	0.60	0.42
3	*k*-NN	Elec/Morgan	V_{OC}	0.65	0.61	0.44
3	*k*-NN	Elec/Morgan	J_{SC}	0.55	0.58	0.41
3	*k*-NN	Elec/Morgan	η	0.61	0.62	0.45
4	KRR	Elec	η	0.49	0.49	0.35
4	KRR	Daylight	η	0.43	0.39	0.28
4	KRR	Morgan	η	0.57	0.58	0.42
4	KRR	Elec/Daylight	η	0.59	0.57	0.40
4	**KRR**	**Elec/Morgan**	$\boldsymbol{\eta}$	**0.68**	**0.70**	**0.51**

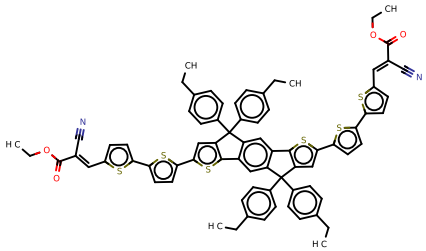
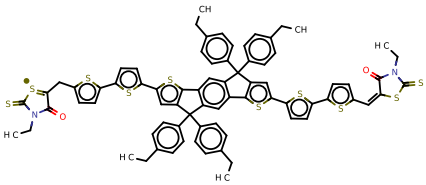
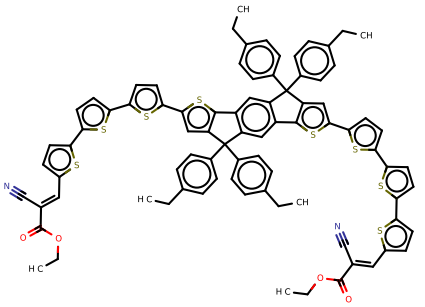
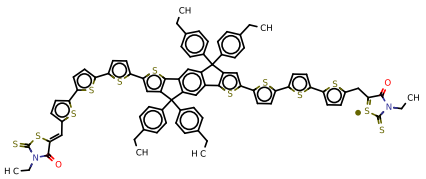
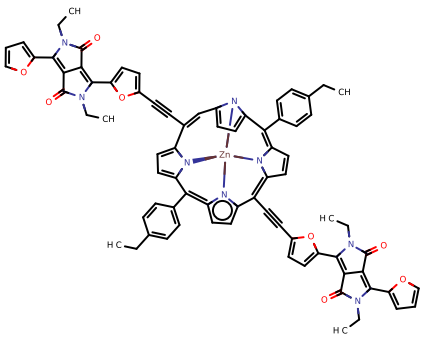
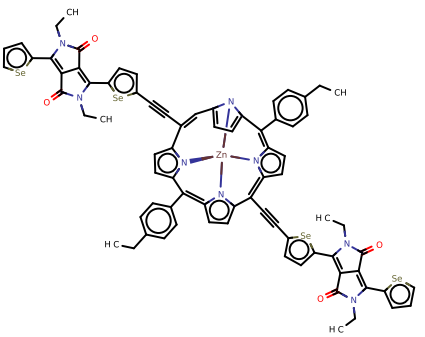
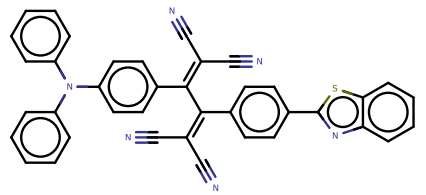
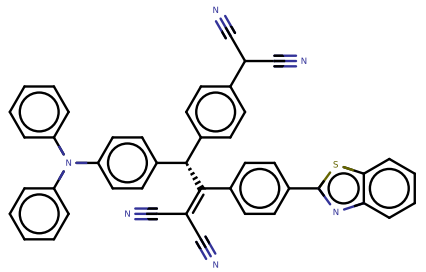
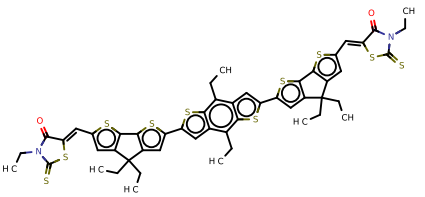
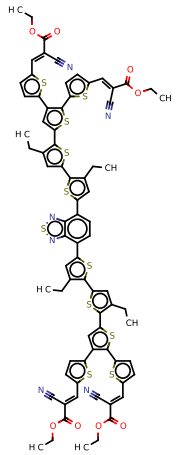
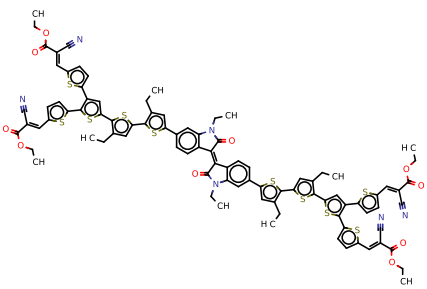
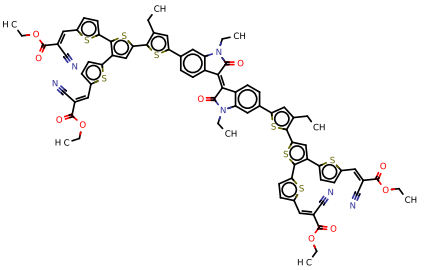
Table S3: Evaluation of features importance by Feature Elimination in KRR models based on Electronic Distance Kernel.

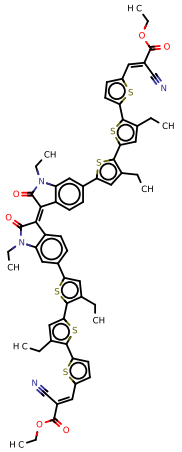
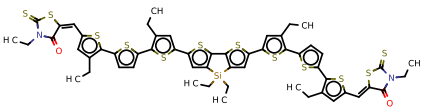
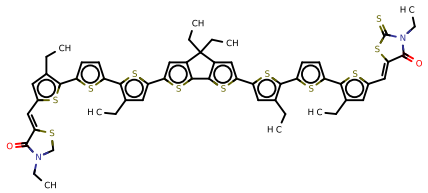
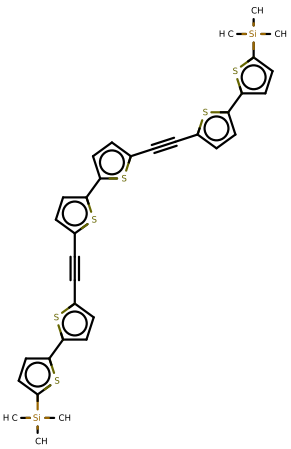
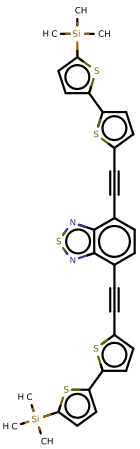
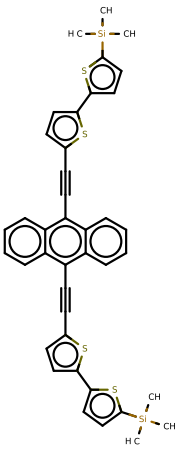
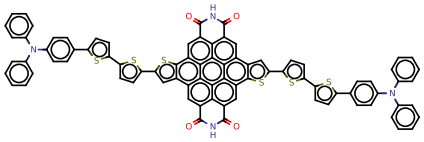
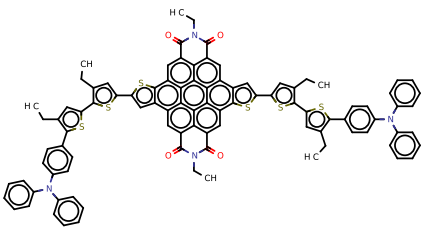
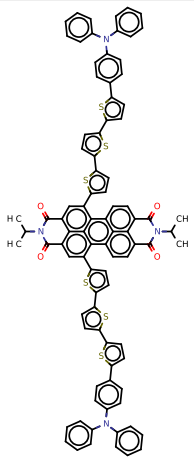
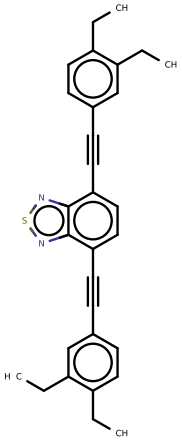
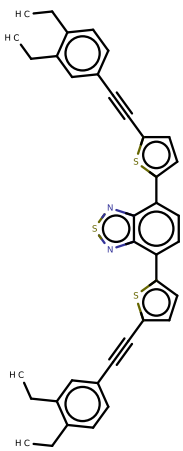
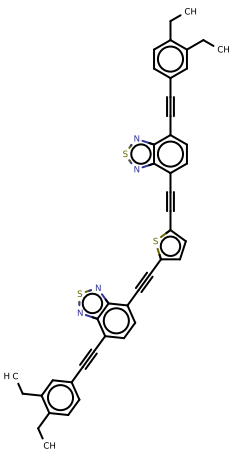
Removed Feature	Model	Input	Target Property	r	RMSE
/	KRR	Elec	η	0.49	1.5575
λ	KRR	Elec	η	0.49	1.5610
E_A^{LUMO}	KRR	Elec	η	0.45	1.5951
E_D^{HOMO}	KRR	Elec	η	0.44	1.6096
E_D^{LUMO}	KRR	Elec	η	0.25	1.7470

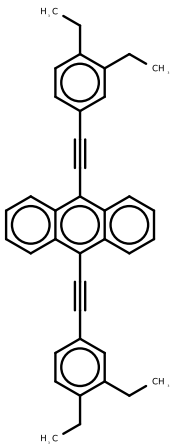
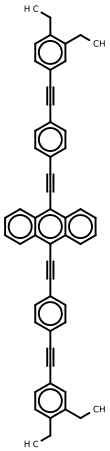
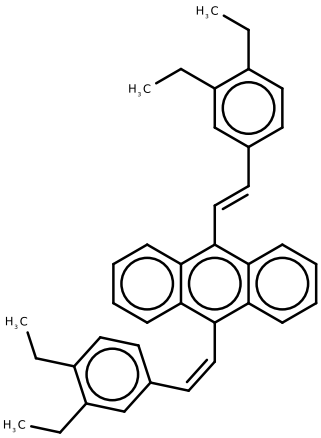
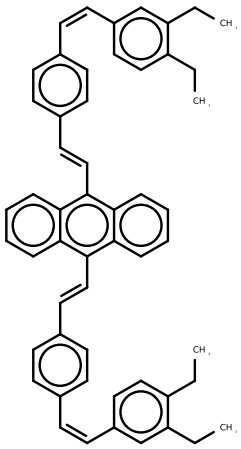
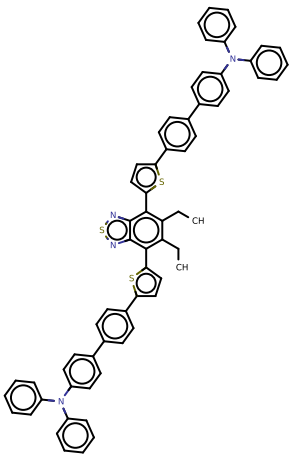
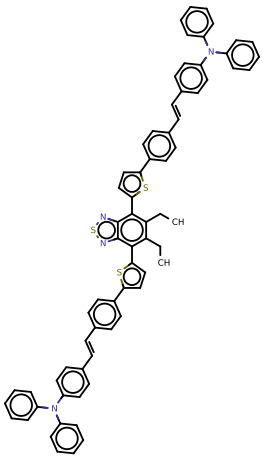
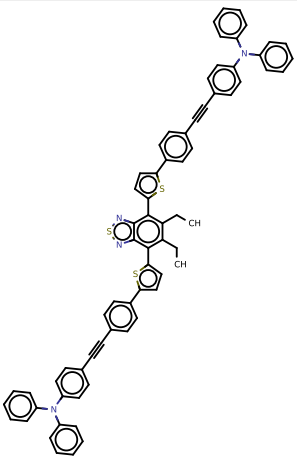
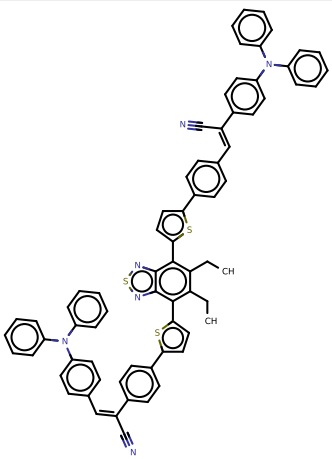
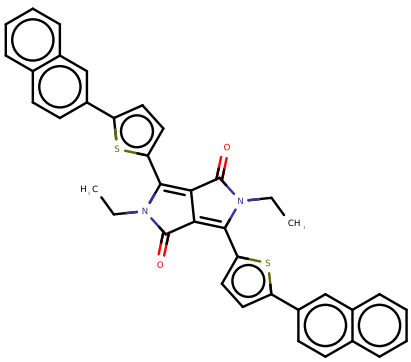
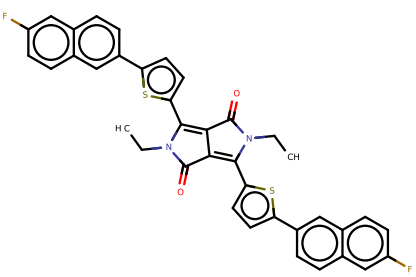
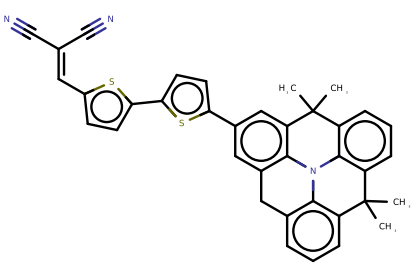
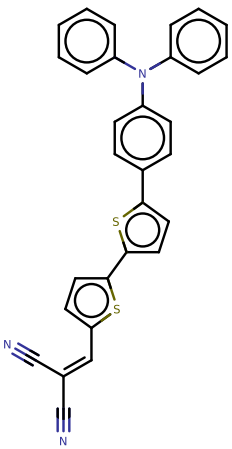
References

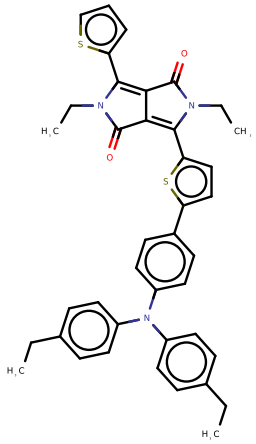
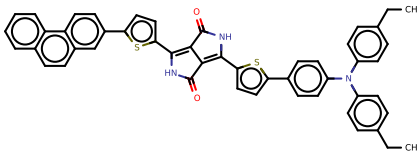
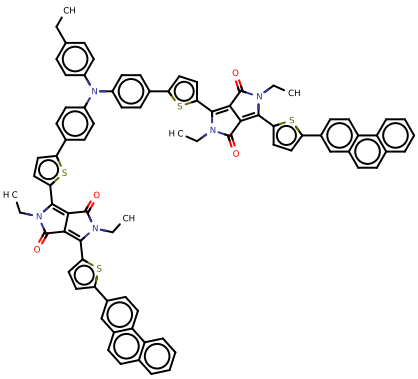
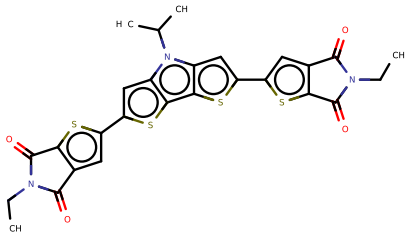
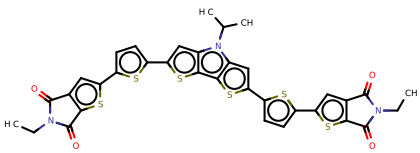
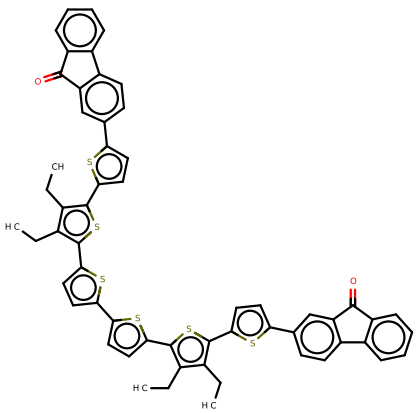
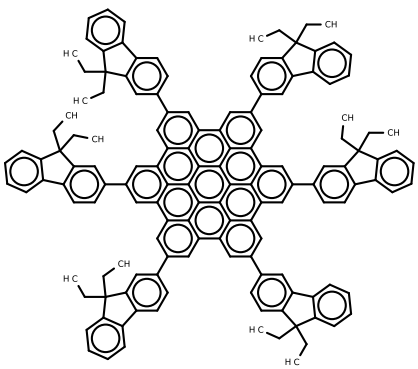
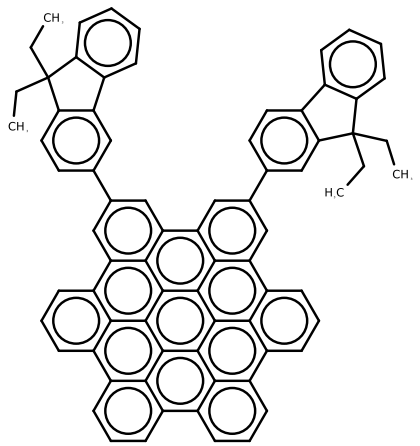
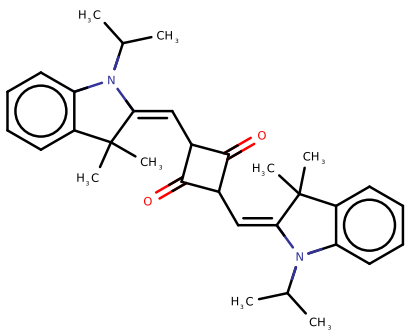
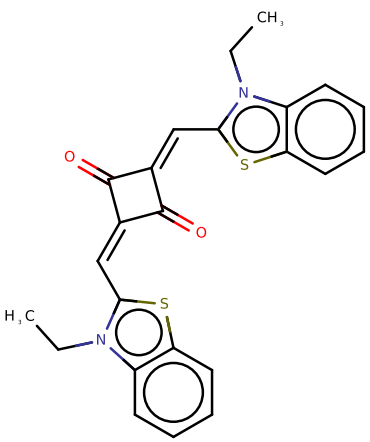
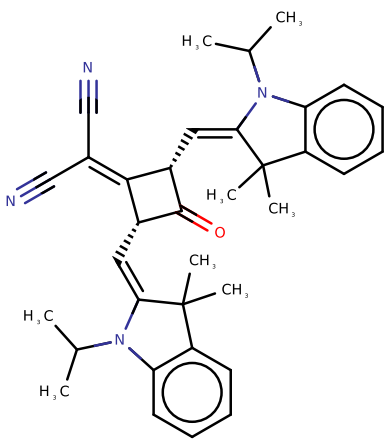
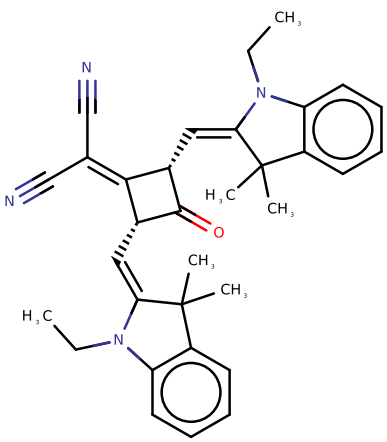
1. Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Petersson, G. A.; Nakatsuji, H.; Li, X.; Caricato, M.; Marenich, A. V.; Bloino, J.; Janesko, B. G.; Gomperts, R.; Mennucci, B.; Hratchian, H. P.; Ortiz, J. V.; Izmaylov, A. F.; Sonnenberg, J. L.; Williams-Young, D.; Ding, F.; Lipparini, F.; Egidi, F.; Goings, J. J.; Peng, B.; Petrone, A.; Henderson, T.; Ranasinghe, D.; Zakrzewski, V. G.; Gao, J.; Rega, N.; Zheng, G.; Liang, W.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Throssell, K.; Montgomery, J. A., Jr.; Peralta, J. E.; Ogliaro, F.; Bearpark, M. J.; Heyd, J. J.; Brothers, E. N.; Kudin, K. N.; Staroverov, V. N.; Keith, T. A.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A. P.; Burant, J. C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Millam, J. M.; Klene, M.; Adamo, C.; Cammi, R.; Ochterski, J. W.; Martin, R. L.; Morokuma, K.; Farkas, O.; Foresman, J. B.; Fox, D. J., Gaussian 16, Revision A.01. 2016.
2. Brédas, J.-L.; Beljonne, D.; Coropceanu, V.; Cornil, J., Charge-Transfer and Energy-Transfer Processes in π-Conjugated Oligomers and Polymers: A Molecular Picture. *Chem. Rev.* **2004,** *104*, 4971-5003.
3. O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R., Open Babel: An open chemical toolbox. *J. Cheminformatics* **2011,** *3*, 33.
4. Landrum, G., RDKit: Open-source cheminformatics.
5. Storn, R.; Price, K., Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization* **1997,** *11*, 341-359.
6. Rupp, M., Machine Learning for Quantum Mechanics in a Nutshell. *Int. J. Quantum Chem.* **2015,** *115*, 1058-1073.
7. Vu, K.; Snyder, J. C.; Li, L.; Rupp, M.; Chen, B. F.; Khelif, T.; Müller, K.-R.; Burke, K., Understanding kernel ridge regression: Common behaviors from simple functions to density functionals. *Int. J. Quantum Chem.* **2015,** *115* (16), 1115-1128.

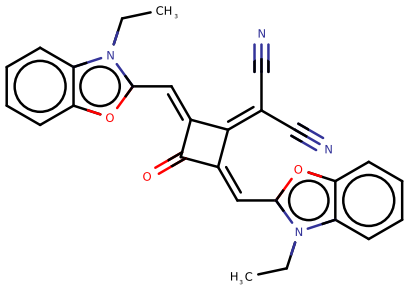
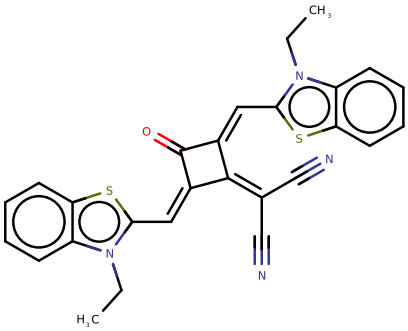
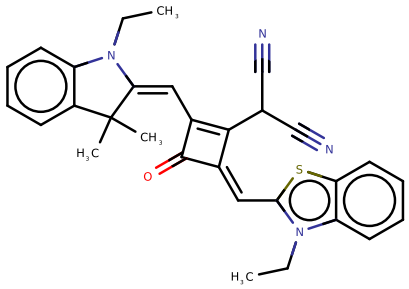
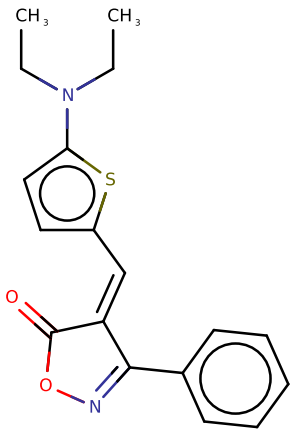
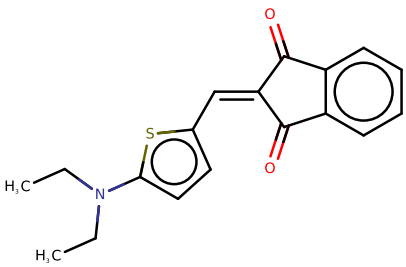
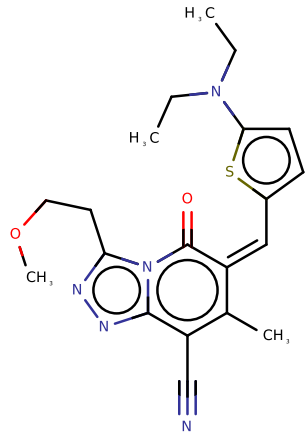
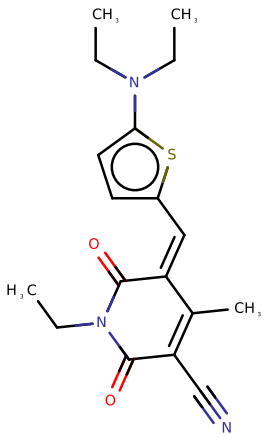
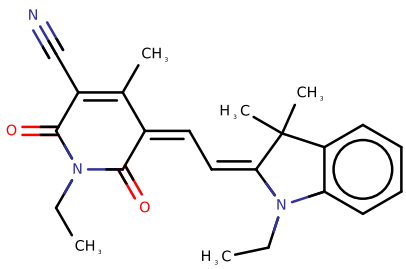
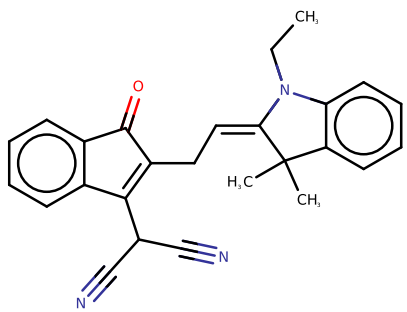
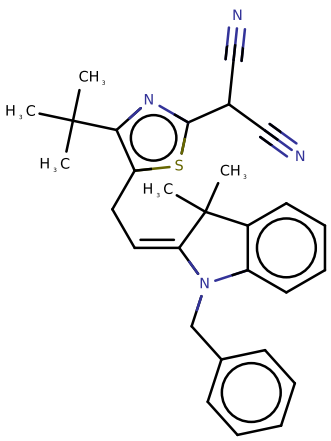
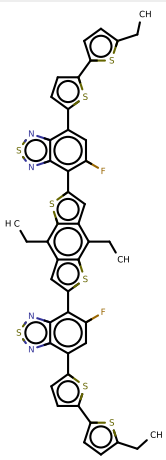
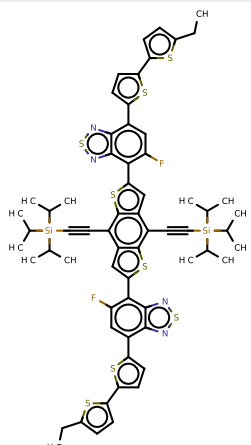
1	2	3
		
4	**5**	**6**
		
7	**8**	**9**
		
10	**11**	**12**
		

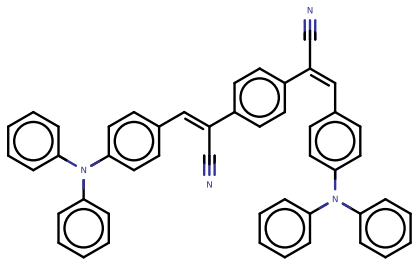
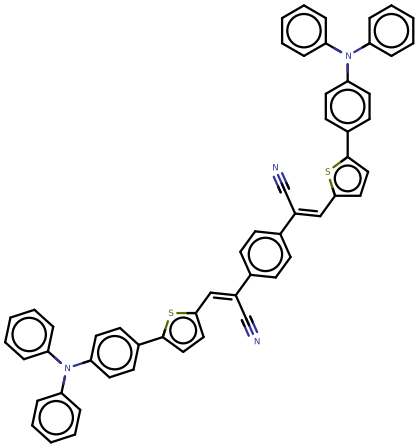
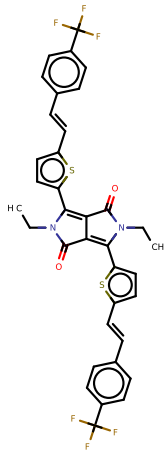
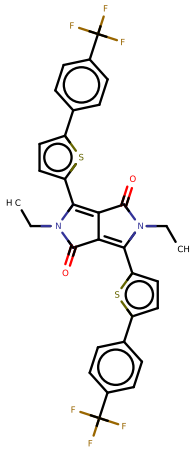
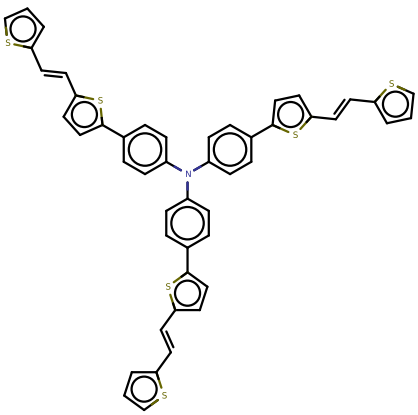
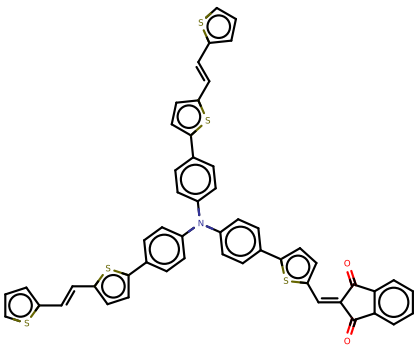
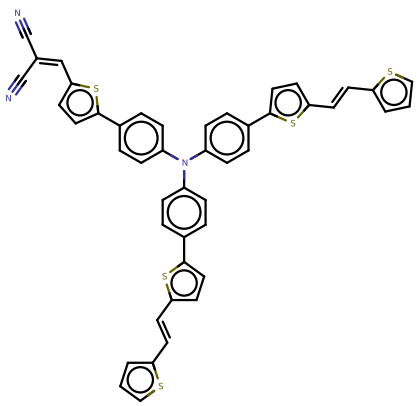
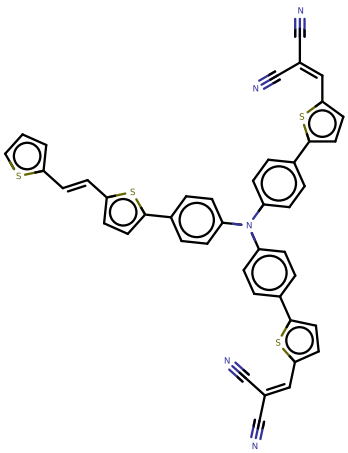
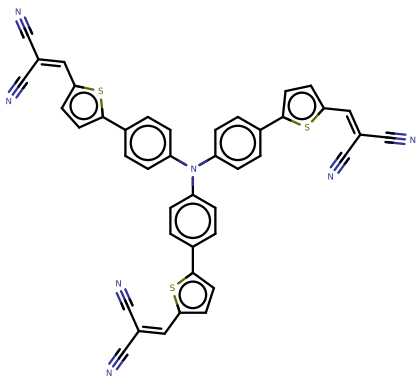
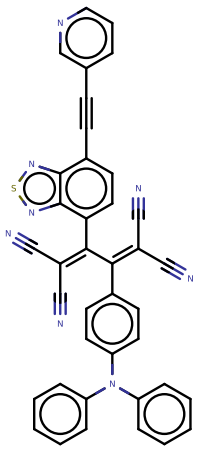
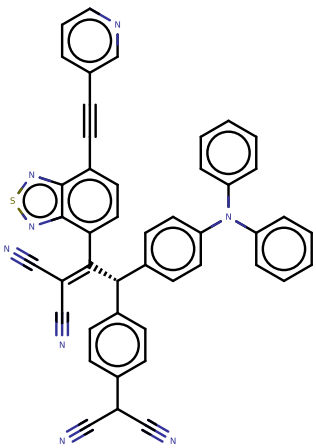
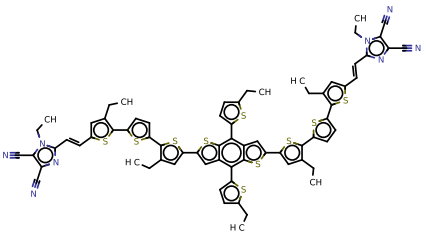
13	14	15
		
16	**17**	**18**
		
19	**20**	**21**
		
22	**23**	**24**
		

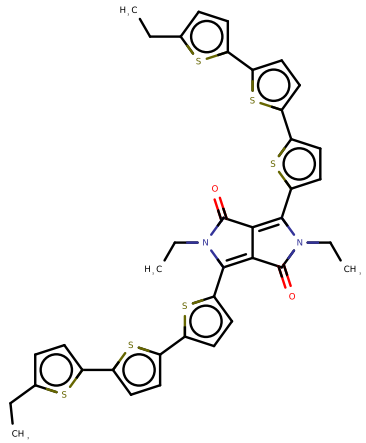
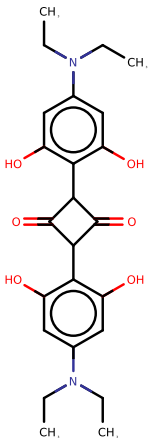
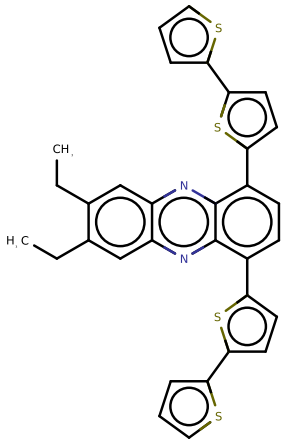
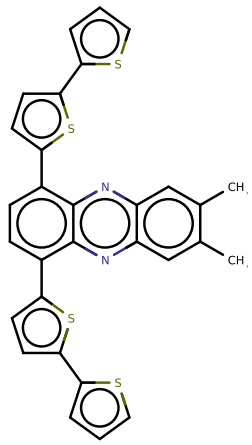
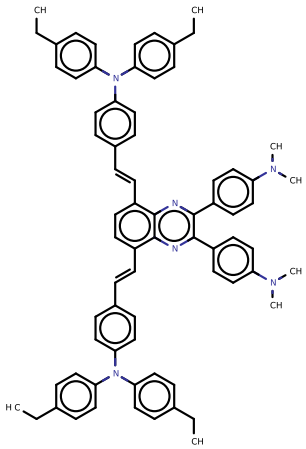
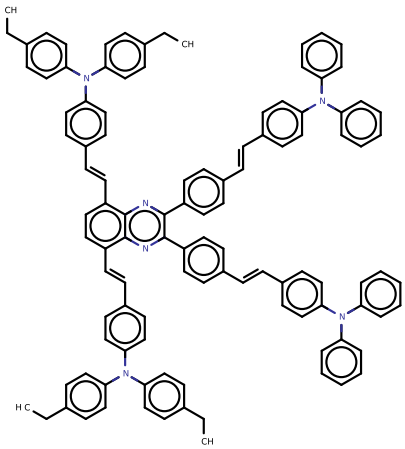
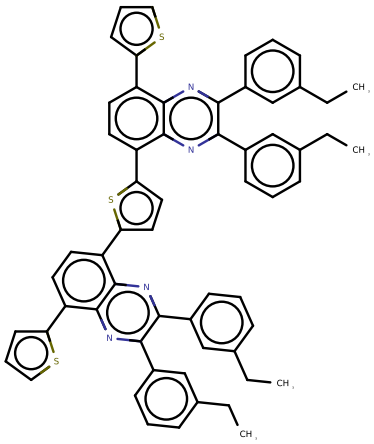
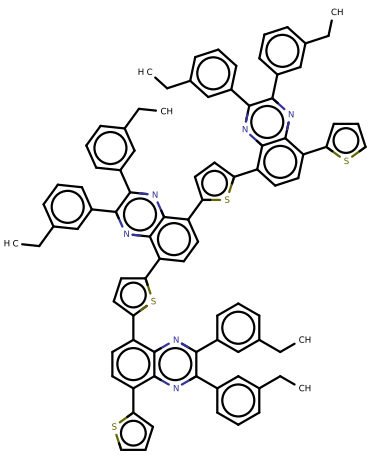
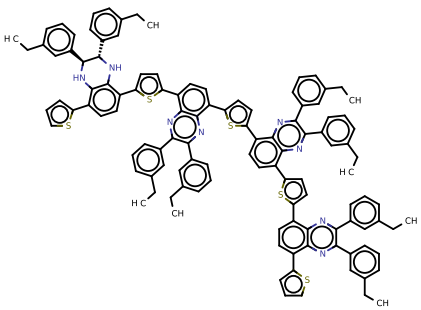
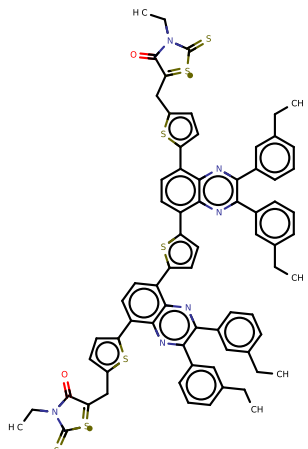
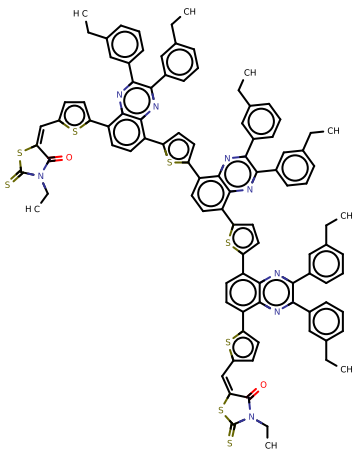
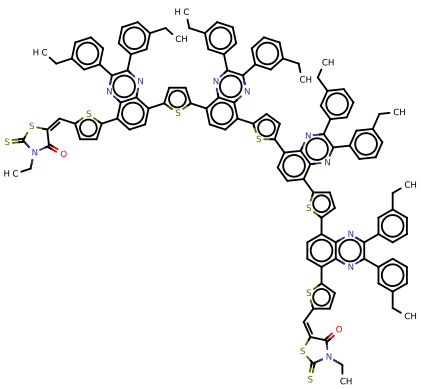
25	26	27
		
28	**29**	**30**
		
31	**32**	**33**
		
34	**35**	**36**
		

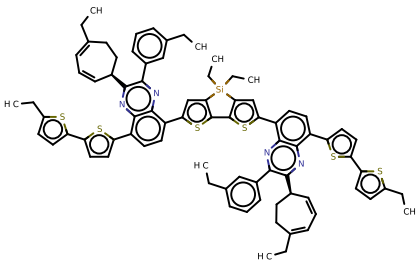
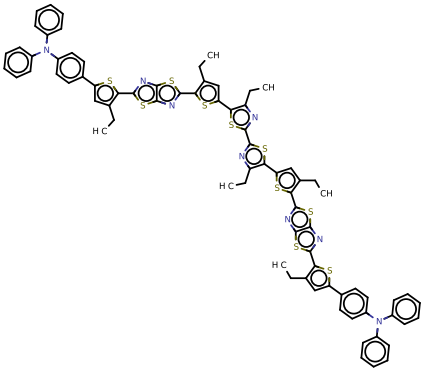
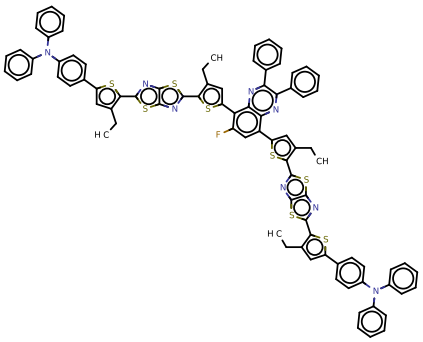
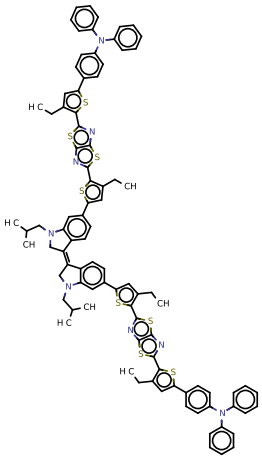
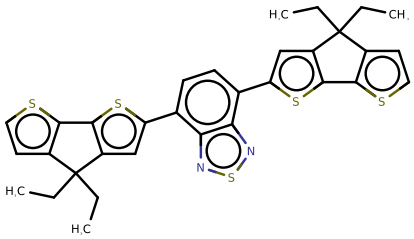
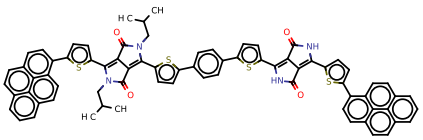
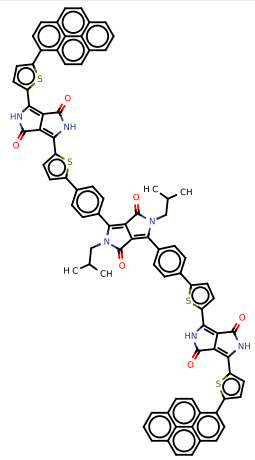
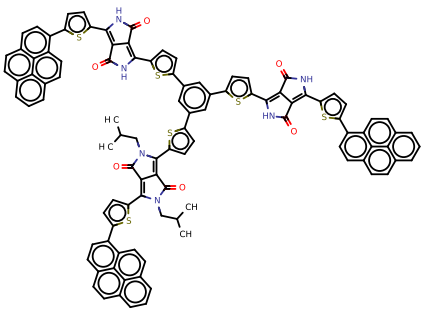
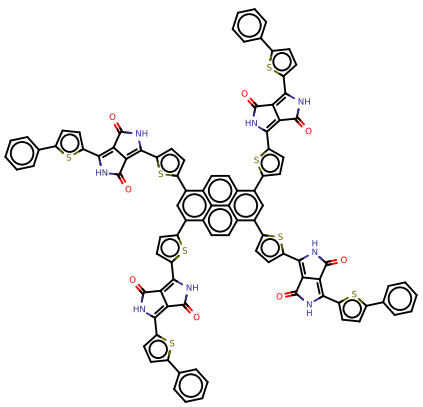
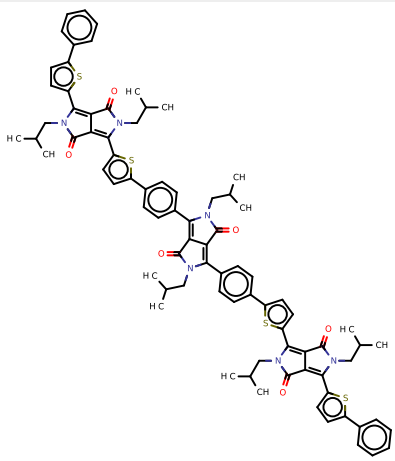
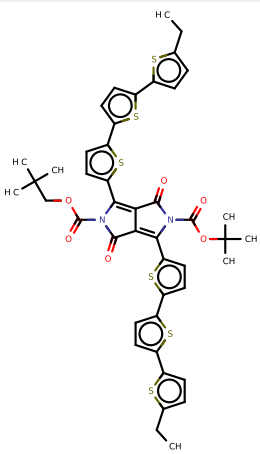
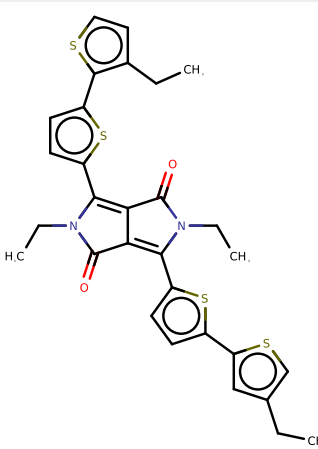
37	38	39
		
40	**41**	**42**
		
43	**44**	**45**
		
46	**47**	**48**
		

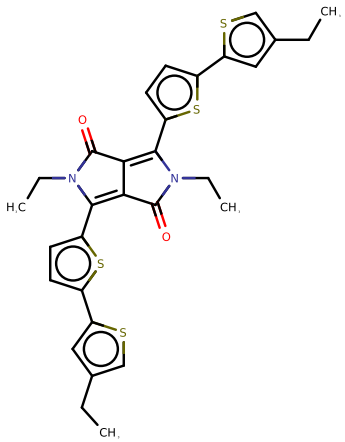
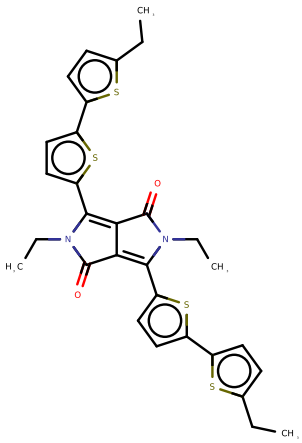
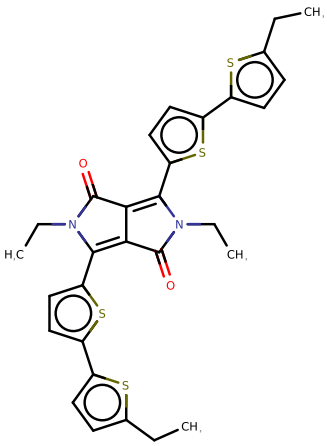
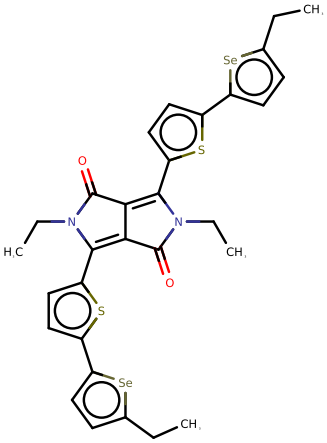
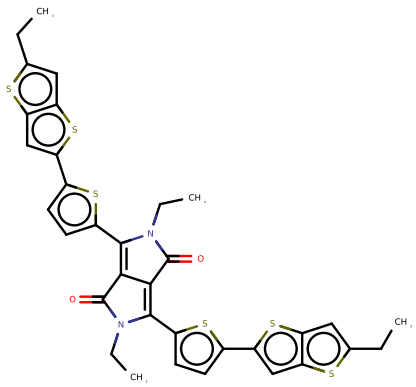
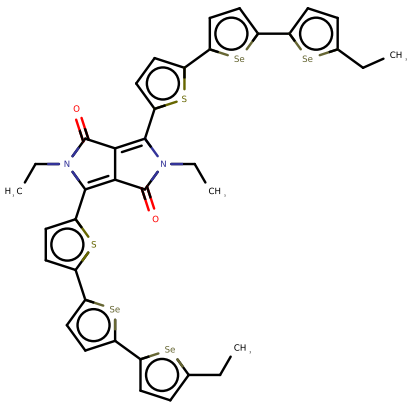
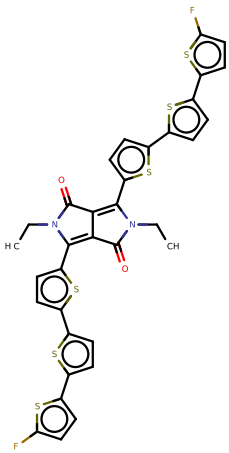
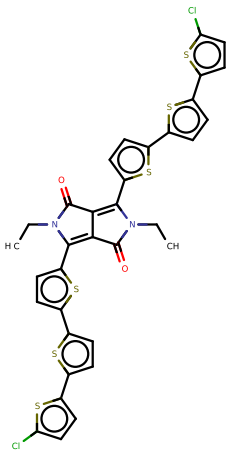
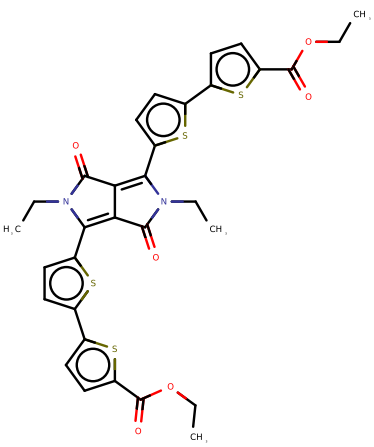
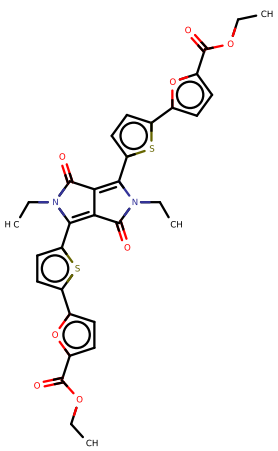
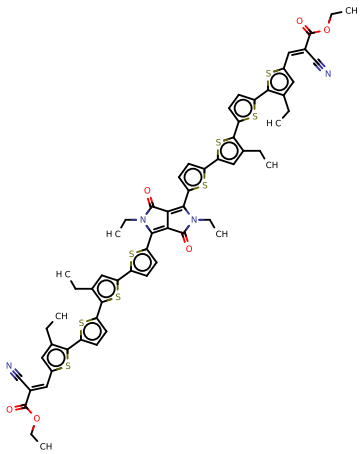
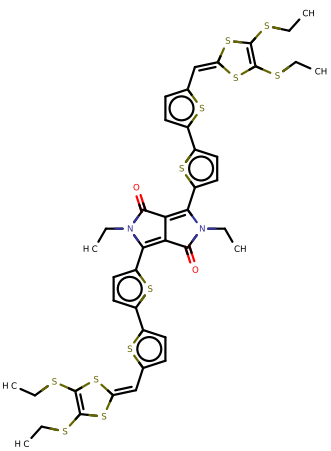
49	50	51
		
52	**53**	**54**
		
55	**56**	**57**
		
58	**59**	**60**
		

61	62	63
		
64	**65**	**66**
		
67	**68**	**69**
		
70	**71**	**72**
		

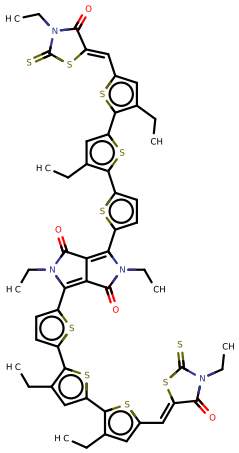
73	74	75
		
76	**77**	**78**
		
79	**80**	**81**
		
82	**83**	**84**
		

85	86	87
		
88	**89**	**90**
		
91	**92**	**93**
		
94	**95**	**96**
		

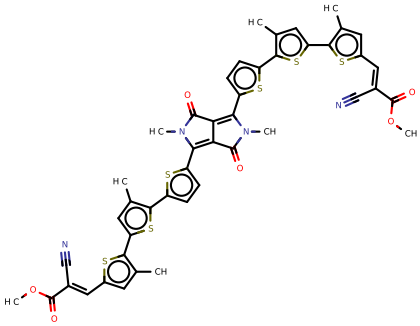
97	98	99
		
100	**101**	**102**
		
103	**104**	**105**
		
106	**107**	**108**
		

109	110	111
		
112	**113**	**114**
		
115	**116**	**117**
		
118	**119**	**120**
		

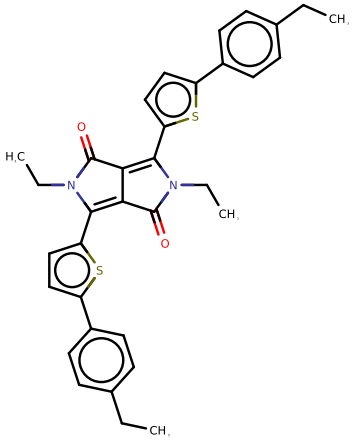
121



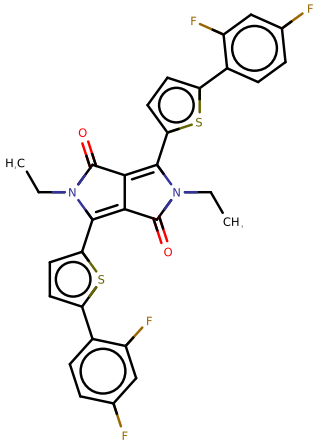
122



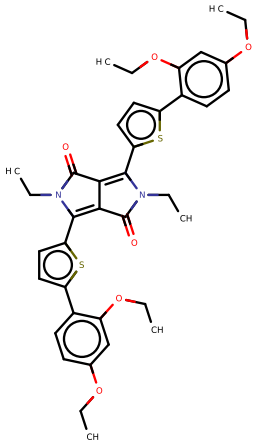
123



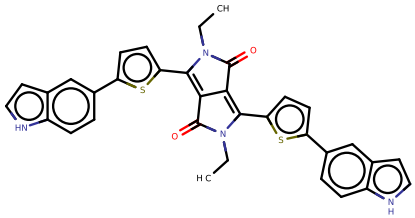
124



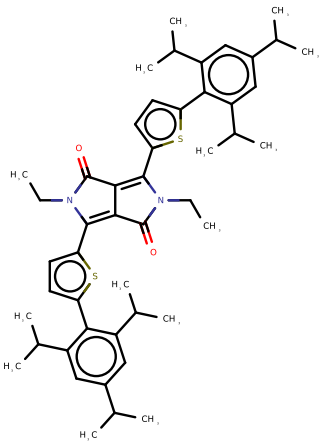
125



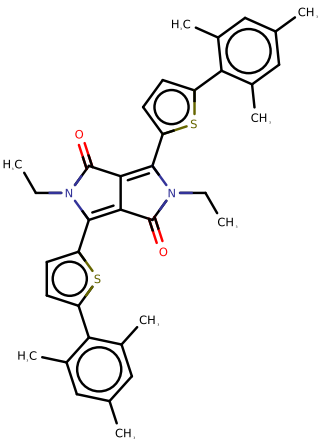
126



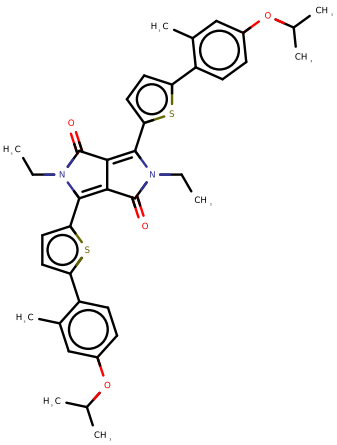
127



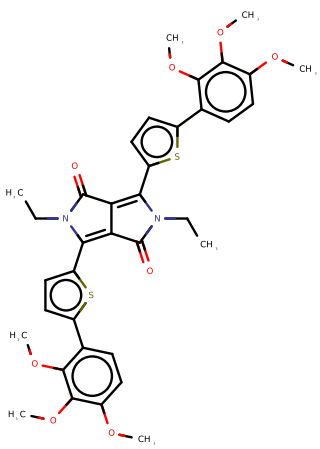
128



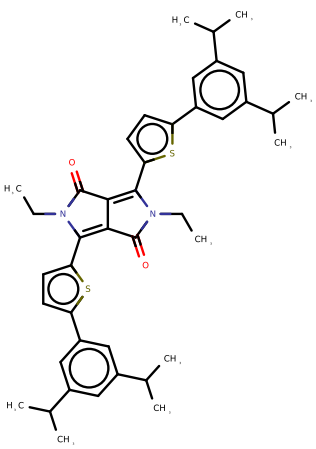
129



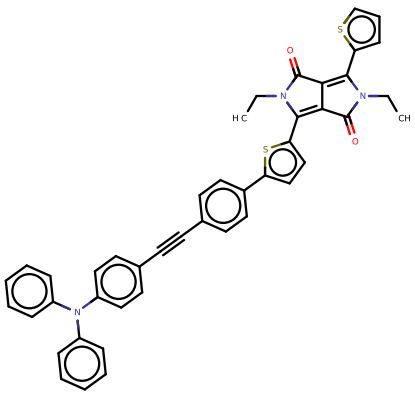
130

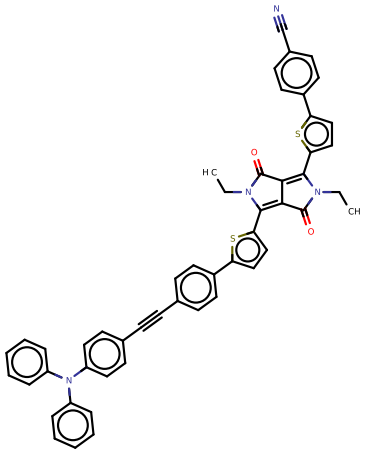
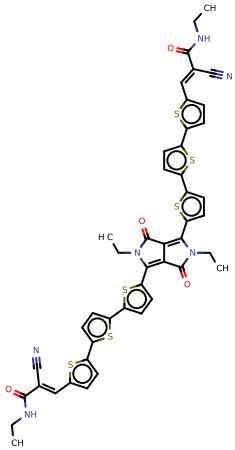
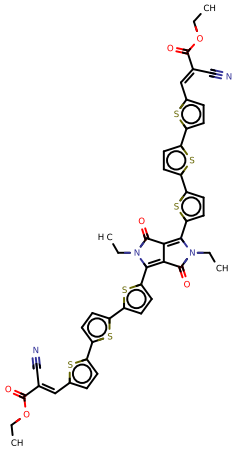
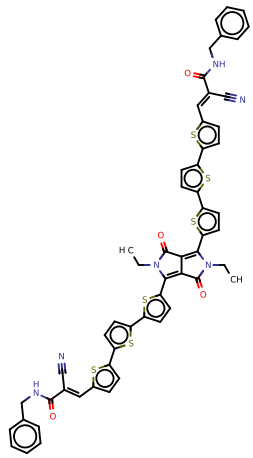
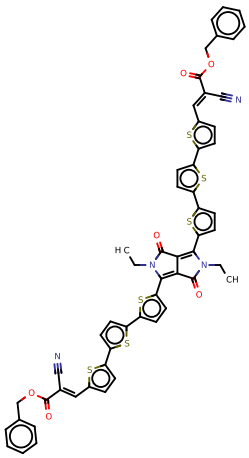
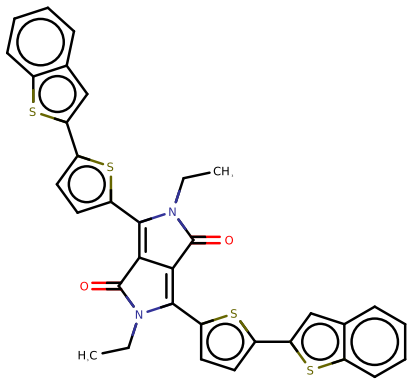
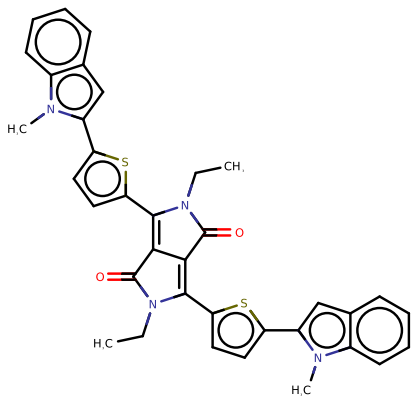
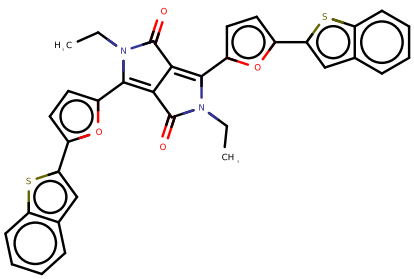
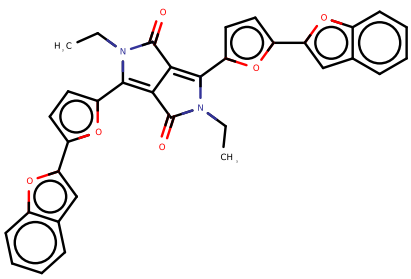
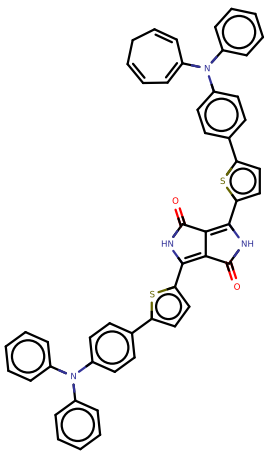
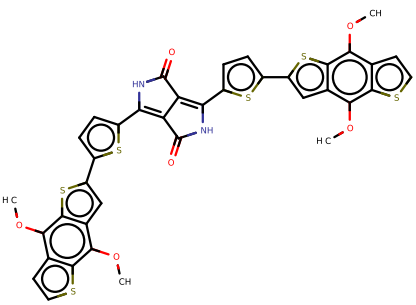
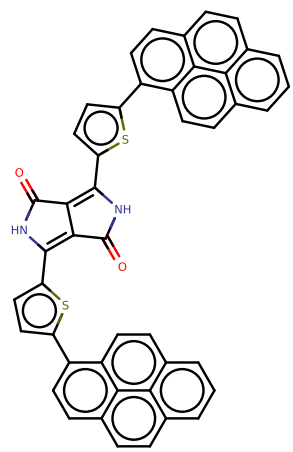


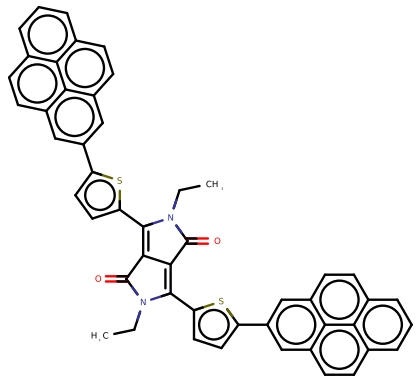
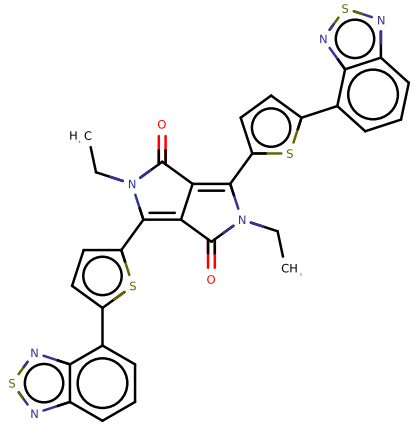
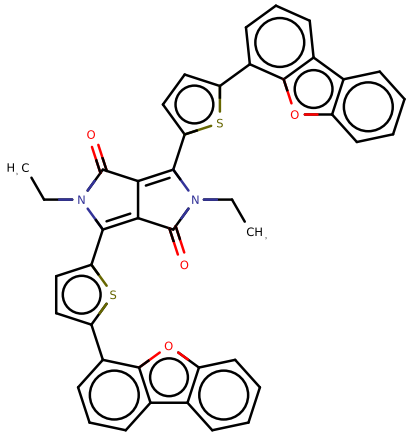
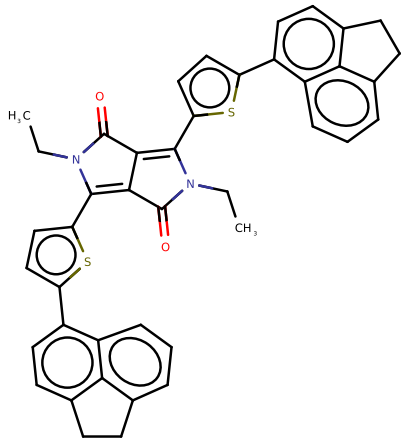
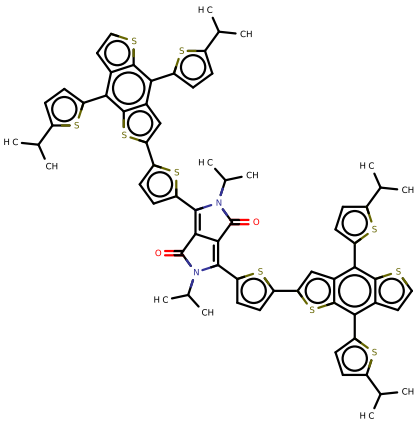
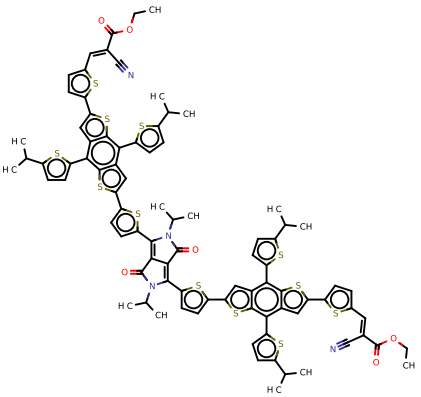
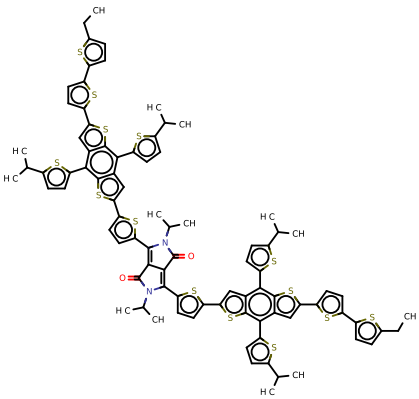
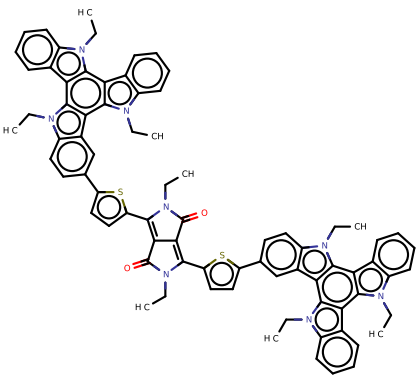
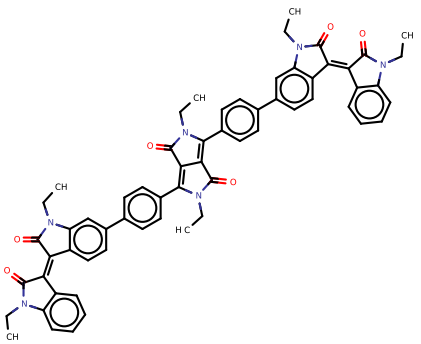
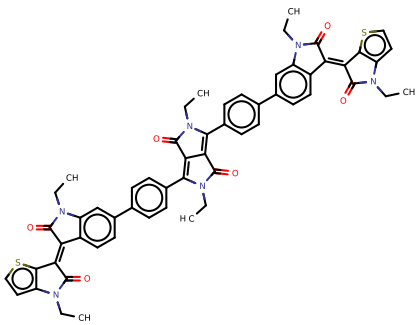
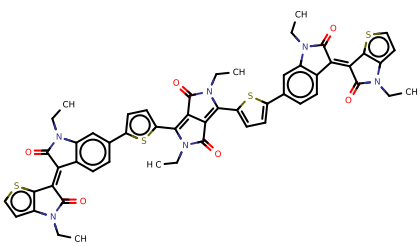
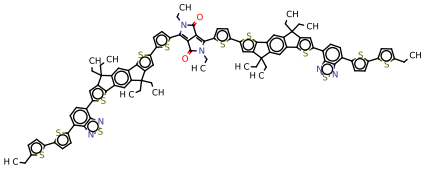
131

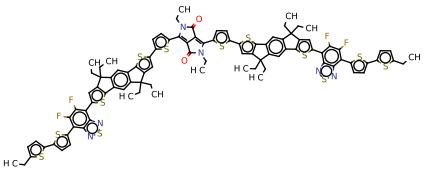
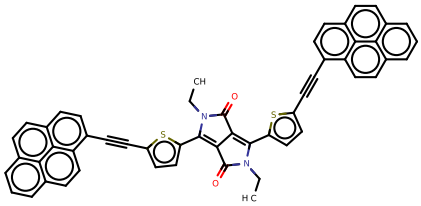
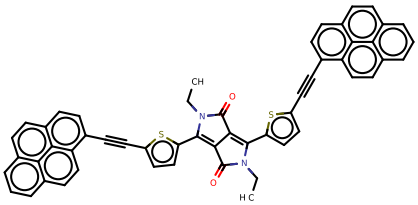
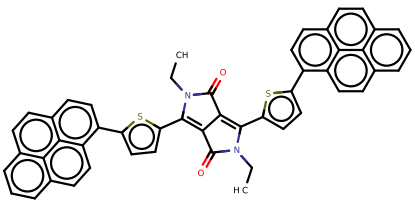
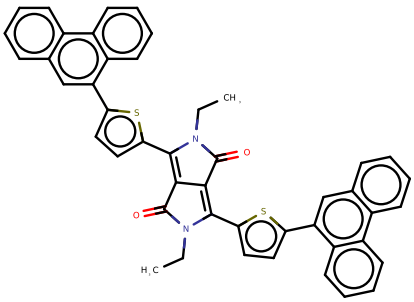
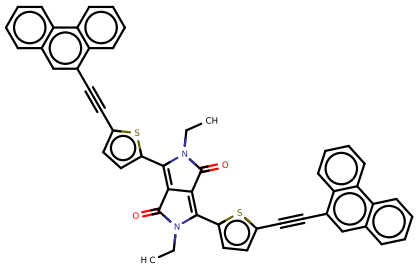
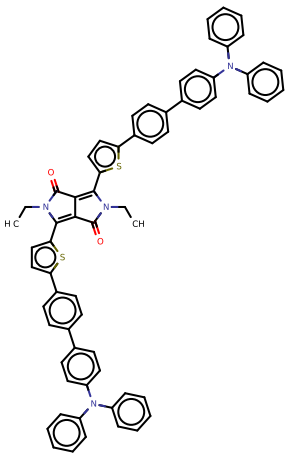
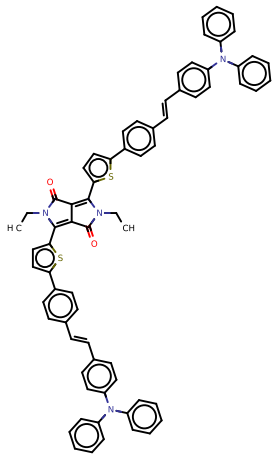
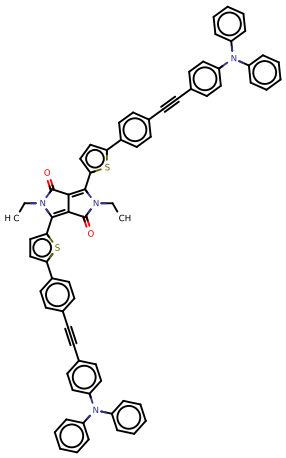
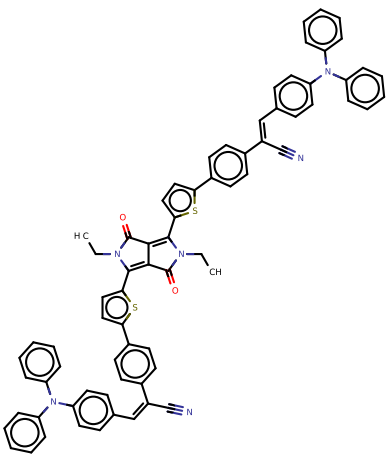
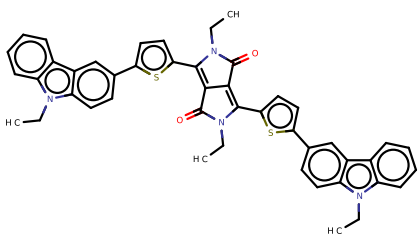
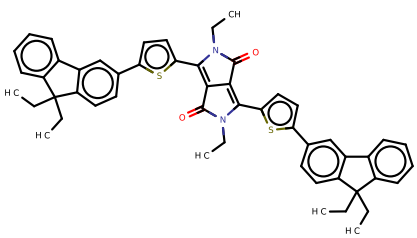


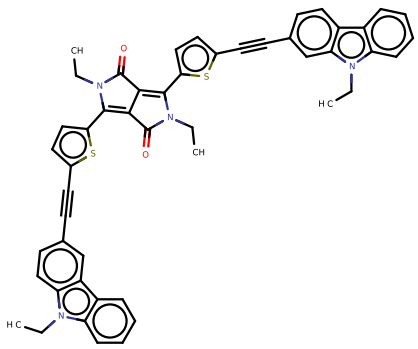
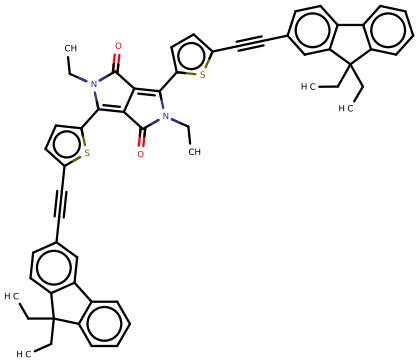
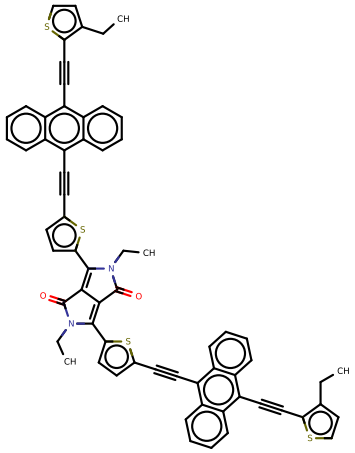
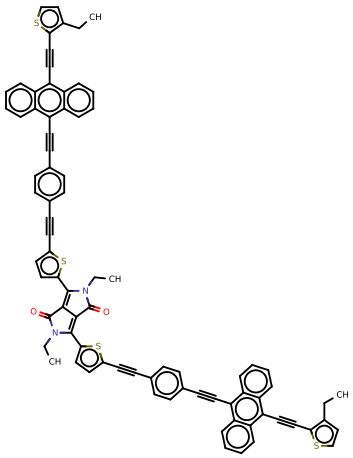
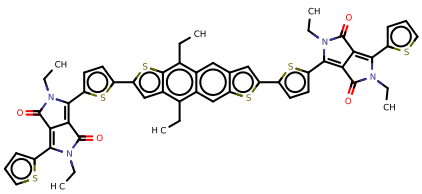
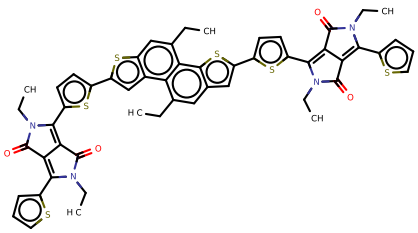
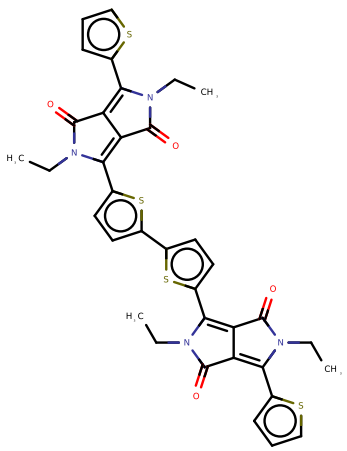
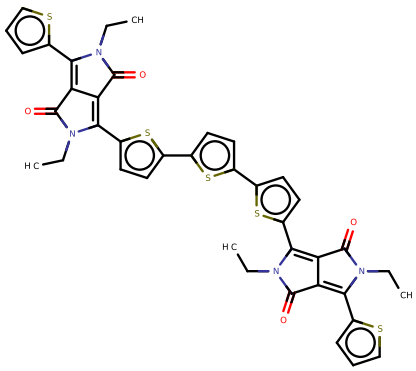
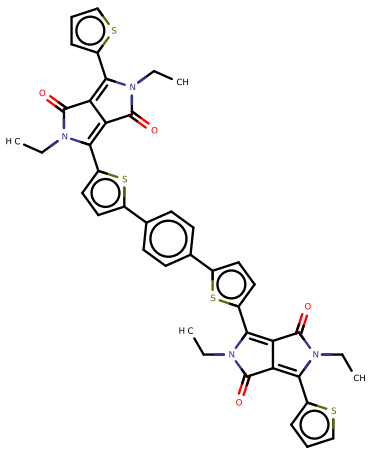
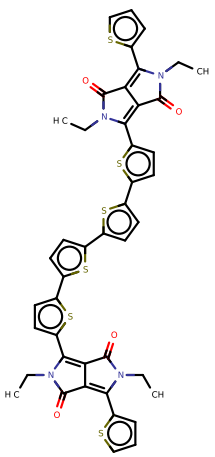
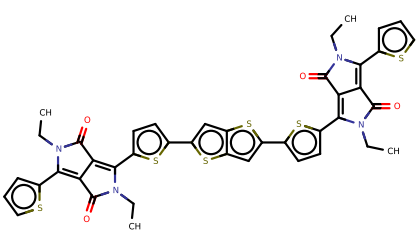
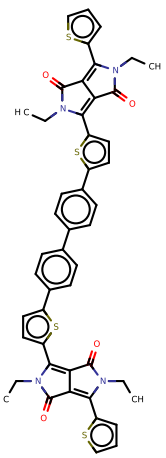
132

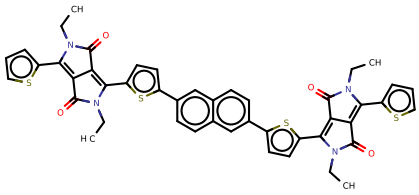
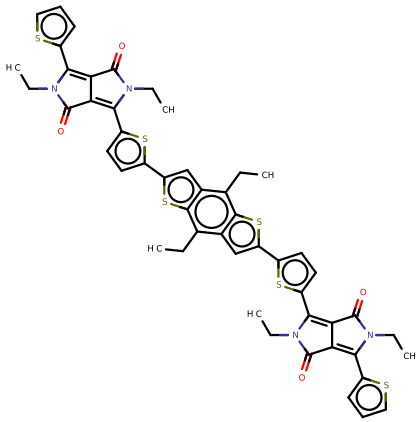
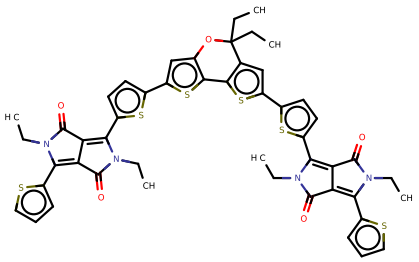
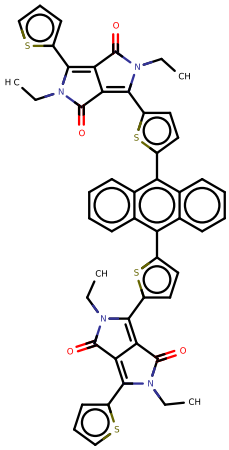
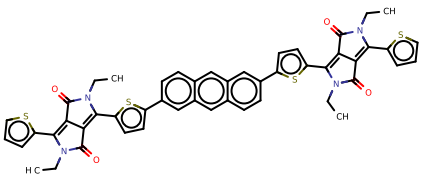
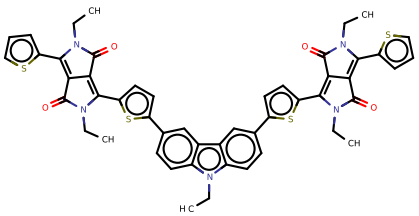
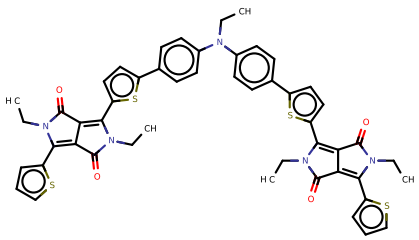
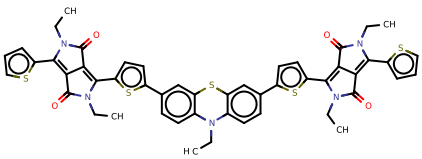
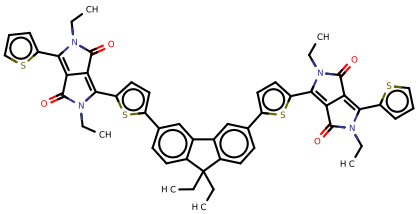
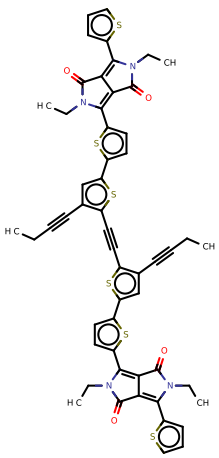
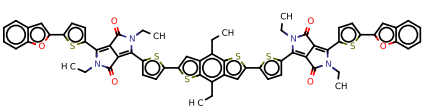
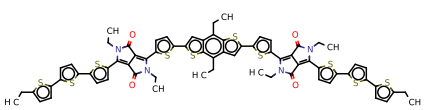


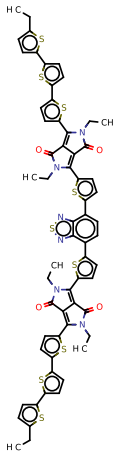
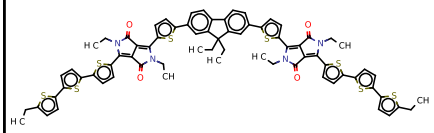
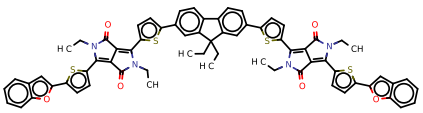
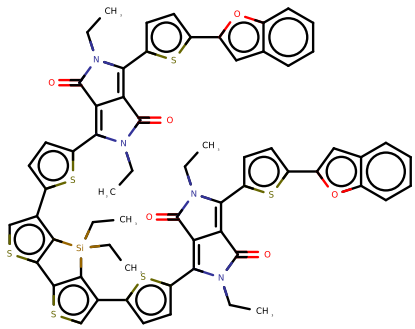
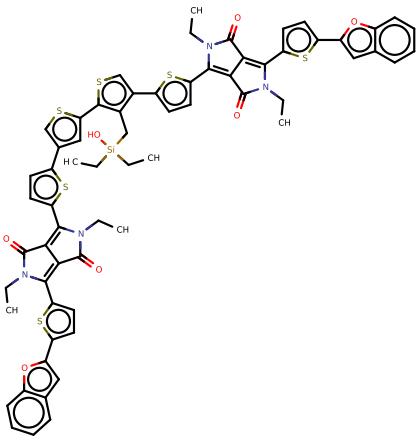
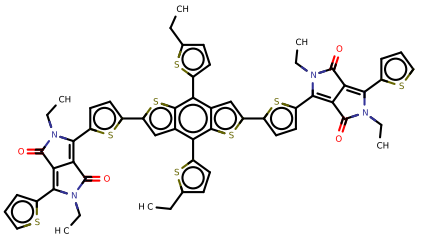
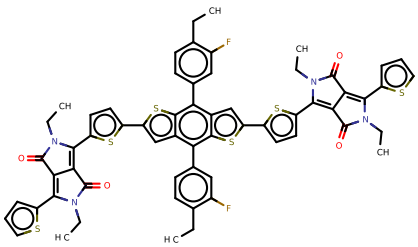
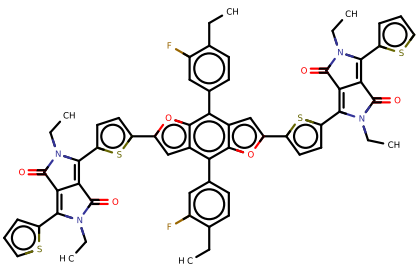
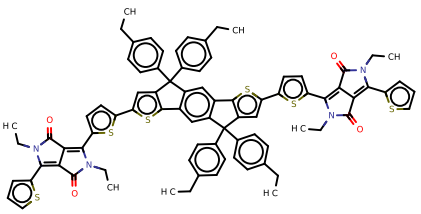
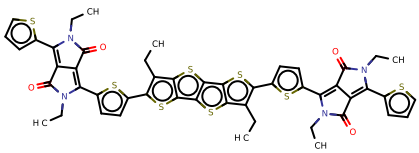
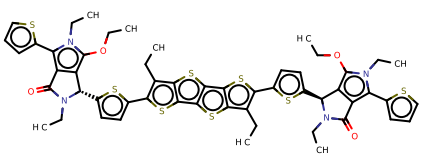
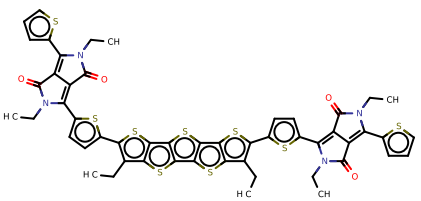
133	134	135
		
136	**137**	**138**
		
139	**140**	**141**
		
142	**143**	**144**
		

145	146	147
		
148	**149**	**150**
		
151	**152**	**153**
		
154	**155**	**156**
		

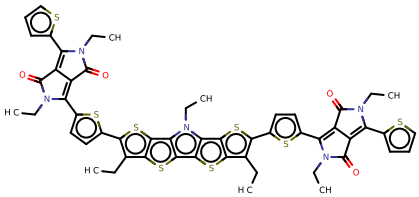
157	158	159
		
160	**161**	**162**
		
163	**164**	**165**
		
166	**167**	**168**
		

169	170	171
		
172	**173**	**174**
		
175	**176**	**177**
		
178	**179**	**180**
		

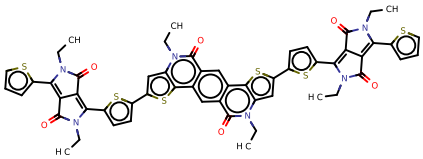
181	182	183
		
184	**185**	**186**
		
187	**188**	**189**
		
190	**191**	**192**
		

193	194	195
		
196	**197**	**198**
		
199	**200**	**201**
		
202	**203**	**204**
		

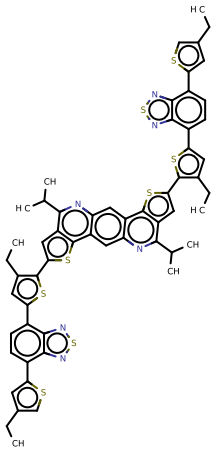
205



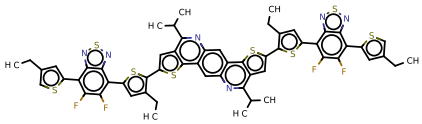
206



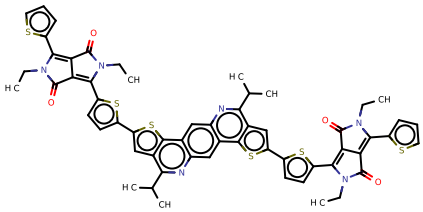
207



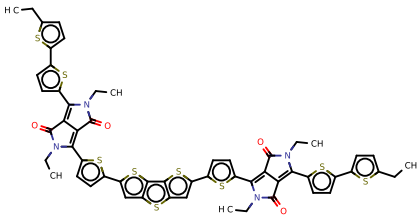
208



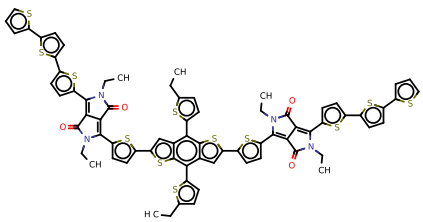
209



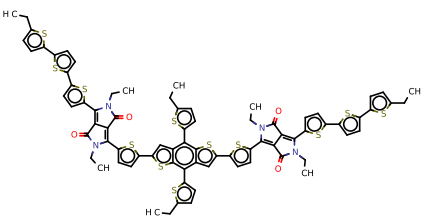
210



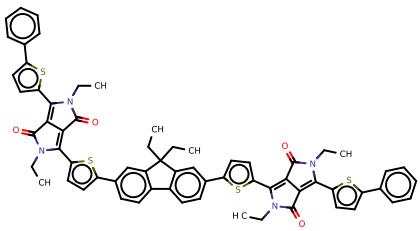
211



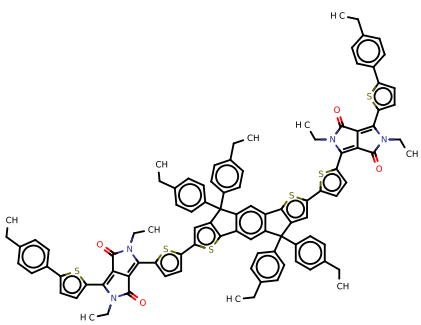
212



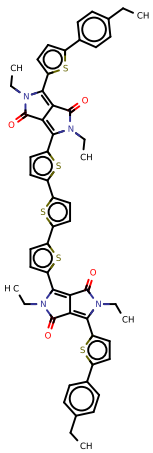
213



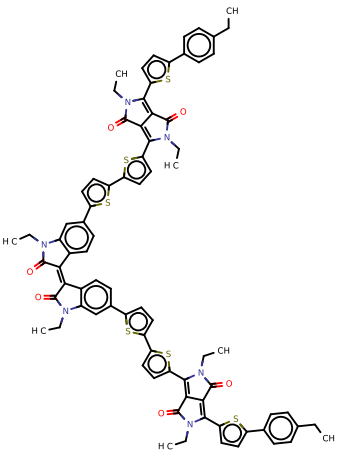
214

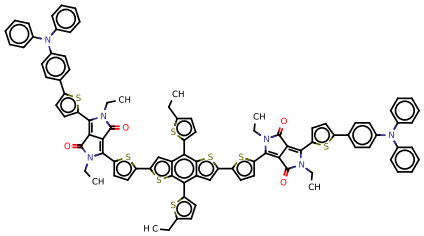
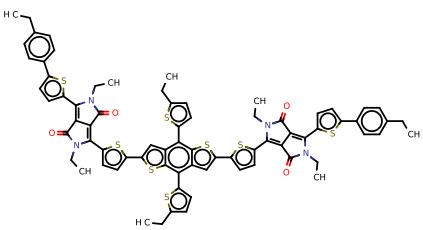
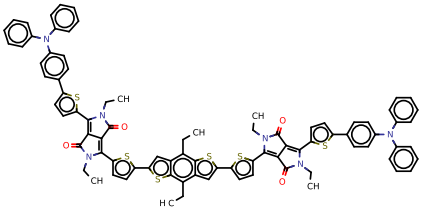
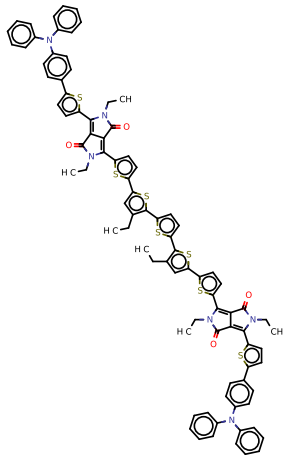
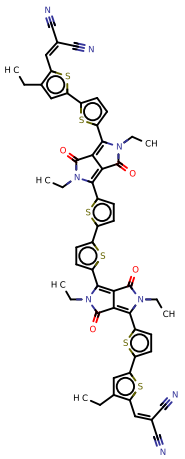
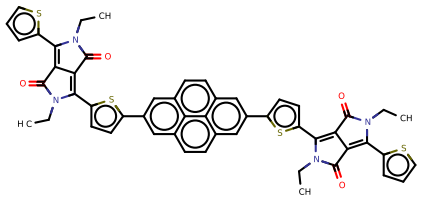
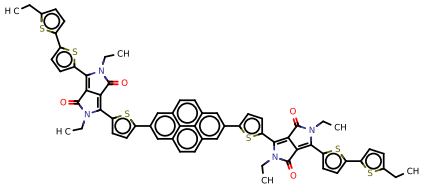
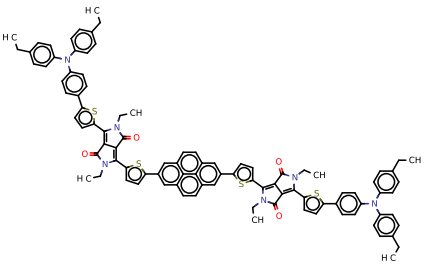
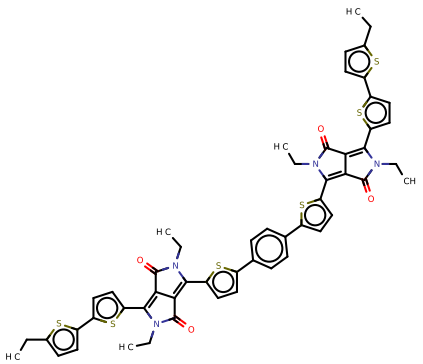
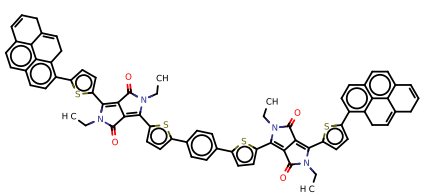
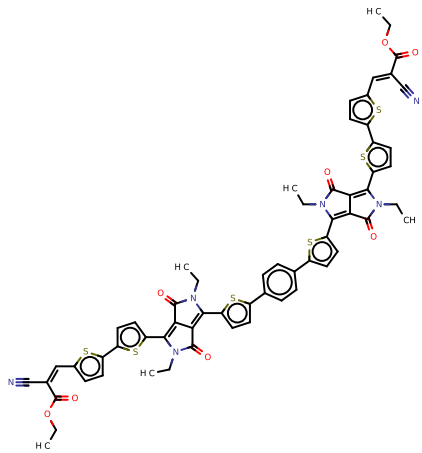
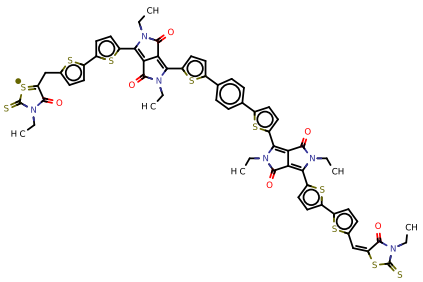


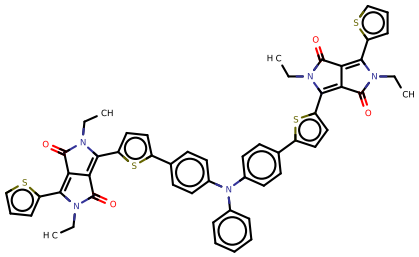
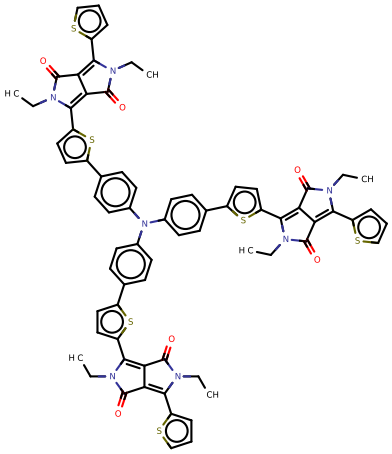
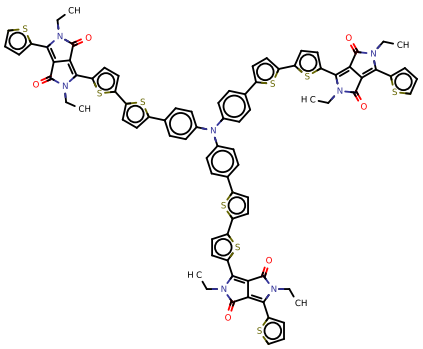
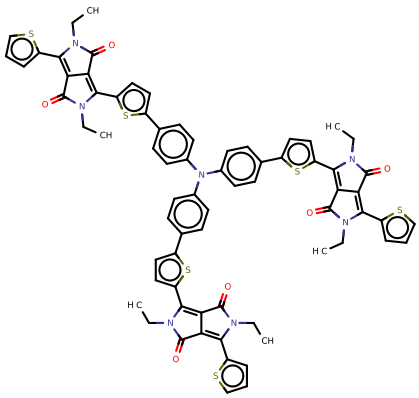
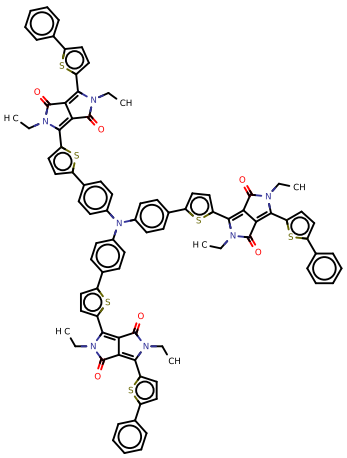
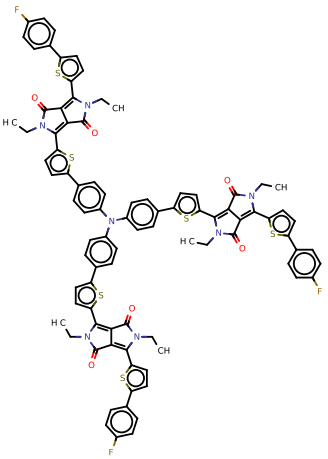
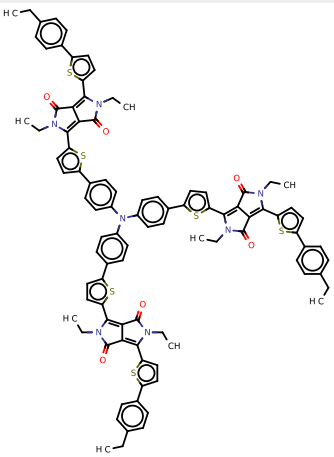
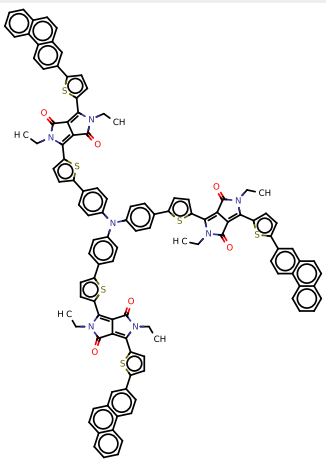
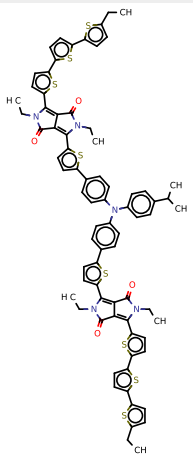
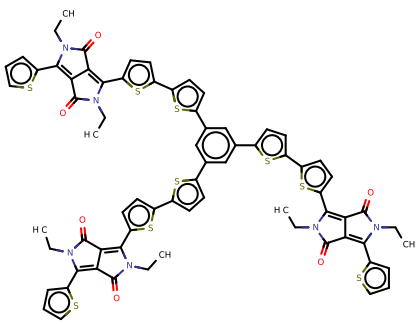
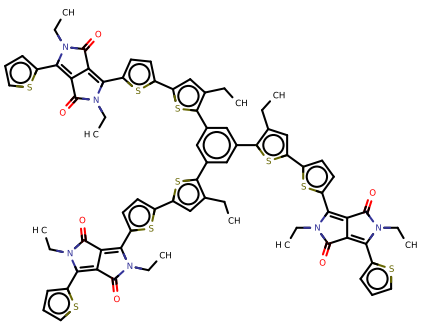
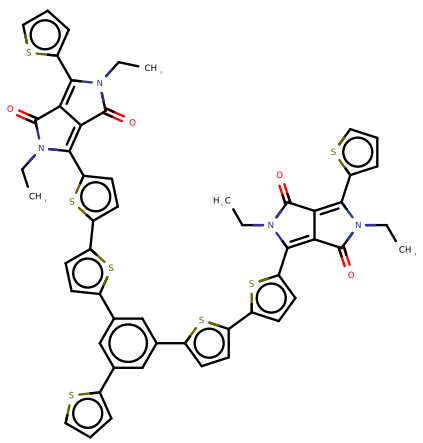
215

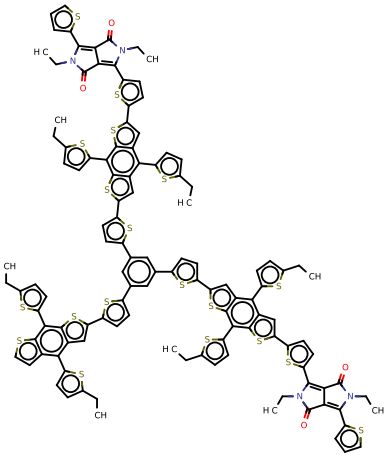
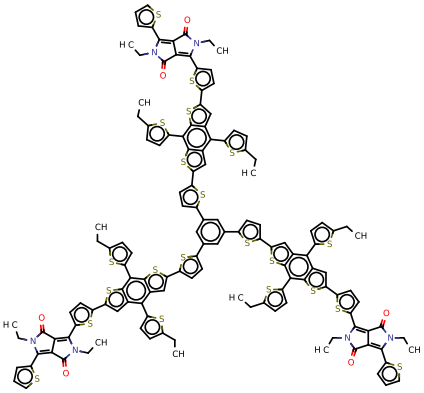
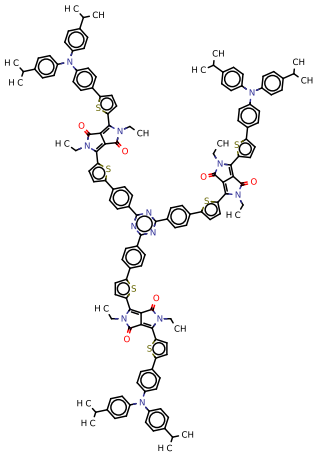
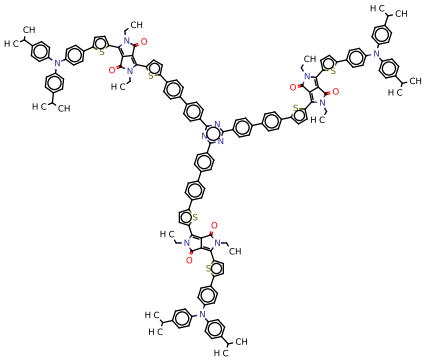
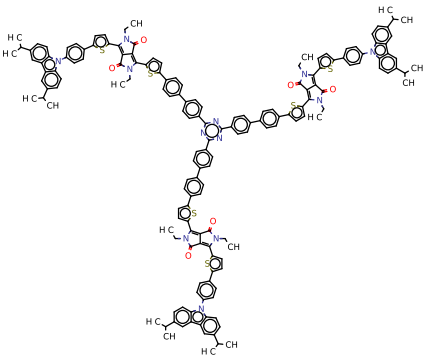
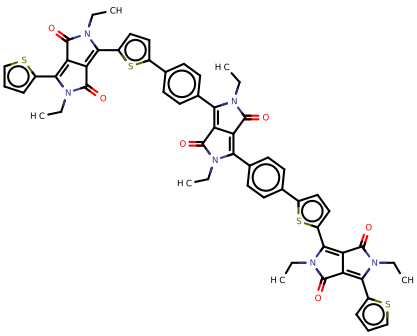


216



217	218	219
		
220	**221**	**222**
		
223	**224**	**225**
		
226	**227**	**228**
		

229	230	231
		
232	**233**	**234**
		
235	**236**	**237**
		
238	**239**	**240**
		

241	242	243
		
244	**245**	**246**
		
247	**248**	**249**