## Setup

```
!pip install chemprop
!pip install rdkit-pypi  # should be included in above after Chemprop v1.6 release

import chemprop
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.offsetbox import AnchoredText
from sklearn.metrics import mean_absolute_error, mean_squared_error
from sklearn.decomposition import PCA
```

```
    Requirement already satisfied: tensorboardX>=2.0 in /usr/local/lib/python3.10/dist-packages (from chemprop) (2.6.
    Requirement already satisfied: torch>=1.4.0 in /usr/local/lib/python3.10/dist-packages (from chemprop) (2.1.0+cu1
    Requirement already satisfied: tqdm>=4.45.0 in /usr/local/lib/python3.10/dist-packages (from chemprop) (4.66.2)
    Requirement already satisfied: typed-argument-parser>=1.6.1 in /usr/local/lib/python3.10/dist-packages (from chem
    Requirement already satisfied: rdkit>=2020.03.1.0 in /usr/local/lib/python3.10/dist-packages (from chemprop) (202
    Requirement already satisfied: Werkzeug>=2.2.2 in /usr/local/lib/python3.10/dist-packages (from flask>=1.1.2->che
    Requirement already satisfied: Jinja2>=3.0 in /usr/local/lib/python3.10/dist-packages (from flask>=1.1.2->chempro
    Requirement already satisfied: itsdangerous>=2.0 in /usr/local/lib/python3.10/dist-packages (from flask>=1.1.2->c
    Requirement already satisfied: click>=8.0 in /usr/local/lib/python3.10/dist-packages (from flask>=1.1.2->chemprop
    Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages (from hyperopt>=0.2.3->chemprop)
    Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from hyperopt>=0.2.3->chemprop) (1
    Requirement already satisfied: networkx>=2.2 in /usr/local/lib/python3.10/dist-packages (from hyperopt>=0.2.3->ch
    Requirement already satisfied: future in /usr/local/lib/python3.10/dist-packages (from hyperopt>=0.2.3->chemprop)
    Requirement already satisfied: cloudpickle in /usr/local/lib/python3.10/dist-packages (from hyperopt>=0.2.3->chem
    Requirement already satisfied: py4j in /usr/local/lib/python3.10/dist-packages (from hyperopt>=0.2.3->chemprop) (
    Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1.
    Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1.3->c
    Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1
    Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1
    Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1.3
    Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1.3->
    Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.1.
    Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=
    Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.0.3->chemp
    Requirement already satisfied: xarray in /usr/local/lib/python3.10/dist-packages (from pandas-flavor>=0.2.0->chem
    Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn>=0.22.
    Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn
    Requirement already satisfied: sphinxcontrib-applehelp in /usr/local/lib/python3.10/dist-packages (from sphinx>=3
    Requirement already satisfied: sphinxcontrib-devhelp in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1
    Requirement already satisfied: sphinxcontrib-jsmath in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.
    Requirement already satisfied: sphinxcontrib-htmlhelp>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from sph
    Requirement already satisfied: sphinxcontrib-serializinghtml>=1.1.5 in /usr/local/lib/python3.10/dist-packages (f
    Requirement already satisfied: sphinxcontrib-qthelp in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.
    Requirement already satisfied: Pygments>=2.0 in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.2->chem
    Requirement already satisfied: docutils<0.19,>=0.14 in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.
    Requirement already satisfied: snowballstemmer>=1.1 in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.
    Requirement already satisfied: babel>=1.3 in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.2->chempro
    Requirement already satisfied: alabaster<0.8,>=0.7 in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.2
    Requirement already satisfied: imagesize in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.2->chemprop
    Requirement already satisfied: requests>=2.5.0 in /usr/local/lib/python3.10/dist-packages (from sphinx>=3.1.2->ch
    Requirement already satisfied: protobuf>=3.20 in /usr/local/lib/python3.10/dist-packages (from tensorboardX>=2.0-
    Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from torch>=1.4.0->chemprop)
    Requirement already satisfied: typing-extensions in /usr/local/lib/python3.10/dist-packages (from torch>=1.4.0->c
    Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages (from torch>=1.4.0->chemprop) (1.
    Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages (from torch>=1.4.0->chemprop) (2
    Requirement already satisfied: triton==2.1.0 in /usr/local/lib/python3.10/dist-packages (from torch>=1.4.0->chemp
    Requirement already satisfied: typing-inspect>=0.7.1 in /usr/local/lib/python3.10/dist-packages (from typed-argum
    Requirement already satisfied: docstring-parser>=0.15 in /usr/local/lib/python3.10/dist-packages (from typed-argu
    Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from Jinja2>=3.0->flas
    Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests
    Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests>=2.5.0->sph
    Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests>=2.5.
    Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests>=2.5.
    Requirement already satisfied: mypy-extensions>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from typing-ins
    Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-packages (from sympy->torch>=1.4.0-
    Requirement already satisfied: rdkit-pypi in /usr/local/lib/python3.10/dist-packages (2022.9.5)
    Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from rdkit-pypi) (1.25.2)
    Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages (from rdkit-pypi) (9.4.0)
```

```
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"
```

```
hiv_df = pd.read_csv("HIV.csv")
hiv_df.head()
```

|   | smiles | activity | HIV_active |
|---|---|---|---|
| **0** | CCC1=[O+][Cu-3]2([O+]=C(CC)C1)[O+]=C(CC)CC(CC)... | CI | 0 |
| **1** | C(=Cc1ccccc1)C1=[O+][Cu-3]2([O+]=C(C=Cc3ccccc3... | CI | 0 |
| **2** | CC(=O)N1c2ccccc2Sc2c1ccc1ccccc21 | CI | 0 |
| **3** | Nc1ccc(C=Cc2ccc(N)cc2S(=O)(=O)O)c(S(=O)(=O)O)c1 | CI | 0 |
| **4** | O=S(=O)(O)CCS(=O)(=O)O | CI | 0 |

Next steps: View recommended plots

```
hiv_df.describe()
```

|   | HIV_active |
|---|---|
| **count** | 41127.000000 |
| **mean** | 0.035086 |
| **std** | 0.184001 |
| **min** | 0.000000 |
| **25%** | 0.000000 |
| **50%** | 0.000000 |
| **75%** | 0.000000 |
| **max** | 1.000000 |

```
unique_values = hiv_df['HIV_active'].unique()
print(f"Unique values in 'HIV_active': {unique_values}")
```

```
    Unique values in 'HIV_active': [0 1]
```

```
unique_values = hiv_df['smiles'].unique()
print(f"Unique values in 'smiles': {unique_values}")
print(f"length of uniqe value: {len(unique_values)}")
```

```
    Unique values in 'smiles': ['CCC1=[O+][Cu-3]2([O+]=C(CC)C1)[O+]=C(CC)CC(CC)=[O+]2'
     'C(=Cc1ccccc1)C1=[O+][Cu-3]2([O+]=C(C=Cc3ccccc3)CC(c3ccccc3)=[O+]2)[O+]=C(c2ccccc2)C1'
     'CC(=O)N1c2ccccc2Sc2c1ccc1ccccc21' ...
     'Cc1ccc(N2C(=O)C3c4[nH]c5ccccc5c4C4CCC(C(C)(C)C)CC4C3C2=O)cc1'
     'Cc1cccc(N2C(=O)C3c4[nH]c5ccccc5c4C4CCC(C(C)(C)C)CC4C3C2=O)c1'
     'CCCCCC=C(c1cc(Cl)c(OC)c(-c2nc(C)no2)c1)c1cc(Cl)c(OC)c(-c2nc(C)no2)c1']
    length of uniqe value: 41127
```

```
# Filter rows where 'your_column' is not equal to 1 or 0
filtered_df = hiv_df[(hiv_df['HIV_active'] != 1) & (hiv_df['HIV_active'] != 0)]
filtered_df
```

| smiles | activity | HIV_active |
|---|---|---|

```
# Filter rows where 'target_column' is equal to 1h
hiv_df_filtered_active = hiv_df[hiv_df['HIV_active'] == 1]
hiv_df_filtered_active
```

| | smiles | activity | HIV_active |
|---|---|---|---|
| **11** | O=C(O)Cc1ccc(SSc2ccc(CC(=O)O)cc2)cc1 | CM | 1 |
| **16** | NNP(=S)(NN)c1ccccc1 | CM | 1 |
| **80** | O=Nc1ccc(O)c(N=O)c1O | CM | 1 |
| **203** | Oc1ccc(Cl)cc1C(c1cc(Cl)ccc1O)C(Cl)(Cl)Cl | CM | 1 |
| **234** | NNC(=O)c1ccccc1SSc1ccccc1C(=O)NN | CM | 1 |
| **...** | ... | ... | ... |
| **41090** | Cc1cn(COCCCOCC(=O)c2ccccc2)c(=O)[nH]c1=O | CM | 1 |
| **41092** | Cc1cn(C2CC3C(COC(CCC[Se]c4ccccc4)N3O)O2)c(=O)[... | CM | 1 |
| **41093** | Cc1cn(C2CC3C(COC(CCCC[Se]c4ccccc4)N3O)O2)c(=O)... | CM | 1 |
| **41098** | Cc1cn(C2CC3C(COC(CC[Se]C#N)N3O)O2)c(=O)[nH]c1=O | CM | 1 |
| **41099** | C[Se]CCC1OCC2OC(n3cc(C)c(=O)[nH]c3=O)CC2N1O | CA | 1 |

1443 rows × 3 columns

Next steps: View recommended plots

```
# Filter rows where 'target_column' is equal to 1h
hiv_df_filtered_inactive = hiv_df[hiv_df['HIV_active'] == 0]
hiv_df_filtered_inactive = hiv_df_filtered_inactive.sample(n=1500, axis=0, replace=True)
hiv_df_filtered_inactive
```

| | smiles | activity | HIV_active |
|---|---|---|---|
| **38106** | C#CCNCC(=O)O | CI | 0 |
| **39778** | CSc1nc(Cl)c2c(n1)Sc1nc3cc4c(cc3n1C2O)OCO4 | CI | 0 |
| **3818** | CN1COc2c(n(C)c(=O)[nH]c2=O)C1 | CI | 0 |
| **18172** | CCOCC=NCC(=O)OCC | CI | 0 |
| **3510** | C=CCn1c(N)c(N=O)c(=O)n(C)c1=O | CI | 0 |
| **...** | ... | ... | ... |
| **16924** | CC(=O)C=Cc1cccc(N=S)c1 | CI | 0 |
| **32148** | O=c1c(OS(=O)(=O)O)c(-c2ccc(OS(=O)(=O)O)cc2OS(=... | CI | 0 |
| **7296** | COCc1c(C)oc2c(C)c3oc(=O)cc(C)c3cc12 | CI | 0 |
| **24101** | COC(OC)c1cccc2c1C(=O)CCC1(CC2)OCCCCO1 | CI | 0 |
| **15156** | Cc1ccc(SCC(=O)C2=C(O)CCCC2=O)cc1 | CI | 0 |

1500 rows × 3 columns

Next steps: View recommended plots

```
hiv_df_sampled = pd.concat([hiv_df_filtered_active, hiv_df_filtered_inactive], axis=0, ignore_index=True)
hiv_df_sampled
```

| | smiles | activity | HIV_active |
|---|---|---|---|
| 0 | O=C(O)Cc1ccc(SSc2ccc(CC(=O)O)cc2)cc1 | CM | 1 |
| 1 | NNP(=S)(NN)c1ccccc1 | CM | 1 |
| 2 | O=Nc1ccc(O)c(N=O)c1O | CM | 1 |
| 3 | Oc1ccc(Cl)cc1C(c1cc(Cl)ccc1O)C(Cl)(Cl)Cl | CM | 1 |
| 4 | NNC(=O)c1ccccc1SSc1ccccc1C(=O)NN | CM | 1 |
| ... | ... | ... | ... |
| 2938 | CC(=O)C=Cc1cccc(N=S)c1 | CI | 0 |
| 2939 | O=c1c(OS(=O)(=O)O)c(-c2ccc(OS(=O)(=O)O)cc2OS(=... | CI | 0 |
| 2940 | COCc1c(C)oc2c(C)c3oc(=O)cc(C)c3cc12 | CI | 0 |
| 2941 | COC(OC)c1cccc2c1C(=O)CCC1(CC2)OCCCCO1 | CI | 0 |
| 2942 | Cc1ccc(SCC(=O)C2=C(O)CCCC2=O)cc1 | CI | 0 |

2943 rows × 3 columns

Next steps: View recommended plots

```
hiv_df_sampled.to_csv('HIV_2.csv', index=False)
# .drop(['activity'], axis=1).
hiv_df_sampled_2 = pd.read_csv("HIV_2.csv")
hiv_df_sampled_2.head()
```

| | smiles | activity | HIV_active |
|---|---|---|---|
| 0 | O=C(O)Cc1ccc(SSc2ccc(CC(=O)O)cc2)cc1 | CM | 1 |
| 1 | NNP(=S)(NN)c1ccccc1 | CM | 1 |
| 2 | O=Nc1ccc(O)c(N=O)c1O | CM | 1 |
| 3 | Oc1ccc(Cl)cc1C(c1cc(Cl)ccc1O)C(Cl)(Cl)Cl | CM | 1 |
| 4 | NNC(=O)c1ccccc1SSc1ccccc1C(=O)NN | CM | 1 |

Next steps: View recommended plots

```
arguments = [
    '--data_path', 'HIV_2.csv',
    '--dataset_type', 'classification',
    '--save_dir', 'test_checkpoints_multimolecule',
    '--epochs', '5',
    '--save_smiles_splits',
    '--quiet',
    '--batch_size', '2048',
    '--ignore_columns', 'activity'
]

args = chemprop.args.TrainArgs().parse_args(arguments)
```

```
mean_score, std_score = chemprop.train.cross_validate(args=args, train_func=chemprop.train.run_training)
```

```
    2943it [00:00, 19049.07it/s]
    100%|██████████| 2943/2943 [00:00<00:00, 150657.69it/s]
    100%|██████████| 2943/2943 [00:00<00:00, 3514.32it/s]
    Fold 0
    0it [00:00, ?it/s]Warning: Repeated SMILES found in data, pickle file of split indices cannot distinguish entries
    1662it [00:00, 229209.00it/s]
      0%|          | 0/5 [00:00<?, ?it/s]
      0%|          | 0/2 [00:00<?, ?it/s]
     50%|█████     | 1/2 [00:09<00:09,  9.96s/it]
    100%|██████████| 2/2 [00:11<00:00,  4.76s/it]

      0%|          | 0/1 [00:00<?, ?it/s]
    100%|██████████| 1/1 [00:00<00:00,  1.41it/s]
     20%|██        | 1/5 [00:11<00:47, 11.89s/it]
      0%|          | 0/2 [00:00<?, ?it/s]
     50%|█████     | 1/2 [00:06<00:06,  6.64s/it]
    100%|██████████| 2/2 [00:07<00:00,  3.01s/it]

      0%|          | 0/1 [00:00<?, ?it/s]
```

```
100%|██████████| 1/1 [00:00<00:00,  2.68it/s]
 40%|████      | 2/5 [00:19<00:28,  9.34s/it]
  0%|          | 0/2 [00:00<?, ?it/s]
 50%|█████     | 1/2 [00:05<00:05,  5.01s/it]
100%|██████████| 2/2 [00:05<00:00,  2.54s/it]

  0%|          | 0/1 [00:00<?, ?it/s]
100%|██████████| 1/1 [00:00<00:00,  1.35it/s]
 60%|██████    | 3/5 [00:26<00:16,  8.14s/it]
  0%|          | 0/2 [00:00<?, ?it/s]
 50%|█████     | 1/2 [00:05<00:05,  5.42s/it]
100%|██████████| 2/2 [00:05<00:00,  2.49s/it]

  0%|          | 0/1 [00:00<?, ?it/s]
100%|██████████| 1/1 [00:00<00:00,  2.53it/s]
 80%|████████  | 4/5 [00:32<00:07,  7.44s/it]
  0%|          | 0/2 [00:00<?, ?it/s]
 50%|█████     | 1/2 [00:05<00:05,  5.55s/it]
100%|██████████| 2/2 [00:06<00:00,  2.61s/it]

  0%|          | 0/1 [00:00<?, ?it/s]
100%|██████████| 1/1 [00:00<00:00,  2.54it/s]
100%|██████████| 5/5 [00:39<00:00,  7.82s/it]
Model 0 best validation auc = 0.546204 on epoch 4
Model 0 test auc = 0.634262
Ensemble test auc = 0.634262
1-fold cross validation
        Seed 0 ==> test auc = 0.634262
Overall test auc = 0.634262 +/- 0.000000
Elapsed time = 0:00:43
```

```
mean_score, std_score
```

```
(0.6342618128332415, 0.0)
```

```
bp_df = pd.read_csv("BBBP.csv")
bp_df.head()
```

| | num | name | p_np | smiles |
|---|---|---|---|---|
| **0** | 1 | Propanolol | 1 | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 |
| **1** | 2 | Terbutylchlorambucil | 1 | C(=O)(OC(C)(C)C)CCCc1ccc(cc1)N(CCCl)CCCl |
| **2** | 3 | 40730 | 1 | c12c3c(N4CCN(C)CC4)c(F)cc1c(c(C(O)=O)cn2C(C)CO... |
| **3** | 4 | 24 | 1 | C1CCN(CC1)Cc1cccc(c1)OCCCNC(=O)C |
| **4** | 5 | cloxacillin | 1 | Cc1onc(c2ccccc2Cl)c1C(=O)N[C@H]3[C@H]4SC(C)(C)... |

Next steps: View recommended plots

```
bp_df.tail()
```

| | num | name | p_np | smiles |
|---|---|---|---|---|
| **2045** | 2049 | licostinel | 1 | C1=C(Cl)C(=C(C2=C1NC(=O)C(N2)=O)[N+](=O)[O-])Cl |
| **2046** | 2050 | ademetionine(adenosyl-methionine) | 1 | [C@H]3([N]2C1=C(C(=NC=N1)N)N=C2)[C@@H]([C@@H](... |
| **2047** | 2051 | mesocarb | 1 | [O+]1=N[N](C=C1[N-]C(NC2=CC=CC=C2)=O)C(CC3=CC=... |
| **2048** | 2052 | | 1 | C1=C(OC)C(=CC2=C1C(=[N+](C(=C2CC)C) |

```
bp_df.drop(['num', 'name', 'p_np'], axis=1).to_csv('BBBP_2.csv', index=False)
```

```
bp_df_2 = pd.read_csv("BBBP_2.csv")
bp_df_2.head()
bp_df_2.tail()
```

| | smiles |
|---|---|
| 0 | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 |
| 1 | C(=O)(OC(C)(C)C)CCCc1ccc(cc1)N(CCCl)CCCl |
| 2 | c12c3c(N4CCN(C)CC4)c(F)cc1c(c(C(O)=O)cn2C(C)CO... |
| 3 | C1CCN(CC1)Cc1cccc(c1)OCCCNC(=O)C |
| 4 | Cc1onc(c2ccccc2Cl)c1C(=O)N[C@H]3[C@H]4SC(C)(C)... |

| | smiles |
|---|---|
| 2045 | C1=C(Cl)C(=C(C2=C1NC(=O)C(N2)=O)[N+](=O)[O-])Cl |
| 2046 | [C@H]3([N]2C1=C(C(=NC=N1)N)N=C2)[C@@H]([C@@H](... |
| 2047 | [O+]1=N[N](C=C1[N-]C(NC2=CC=CC=C2)=O)C(CC3=CC=... |
| 2048 | C1=C(OC)C(=CC2=C1C(=[N+](C(=C2CC)C)[NH-])C3=CC... |
| 2049 | [N+](=NCC(=O)N[C@@H]([C@H](O)C1=CC=C([N+]([O-]... |

```
arguments = [
    '--test_path', 'BBBP_2.csv',
    '--preds_path', 'BBBP_preds.csv',
    '--checkpoint_dir', 'test_checkpoints_multimolecule'
]

args = chemprop.args.PredictArgs().parse_args(arguments)
preds = chemprop.train.make_predictions(args=args)
```

```
Loading training args
Setting molecule featurization parameters to default.
Loading data
2050it [00:00, 243751.19it/s]
100%|██████████| 2050/2050 [00:00<00:00, 145991.63it/s]
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: UserWarning: This DataLoader will cre
  warnings.warn(_create_warning_msg(
Validating SMILES
Test size = 2,039
  0%|          | 0/1 [00:00<?, ?it/s]Loading pretrained parameter "encoder.encoder.0.cached_zero_vector".
Loading pretrained parameter "encoder.encoder.0.W_i.weight".
Loading pretrained parameter "encoder.encoder.0.W_h.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.bias".
Loading pretrained parameter "readout.1.weight".
Loading pretrained parameter "readout.1.bias".
Loading pretrained parameter "readout.4.weight".
Loading pretrained parameter "readout.4.bias".
Moving model to cuda

  0%|          | 0/41 [00:00<?, ?it/s]
  2%|▏         | 1/41 [00:01<01:17,  1.94s/it]
 10%|▉         | 4/41 [00:02<00:15,  2.32it/s]
 17%|█▋        | 7/41 [00:02<00:07,  4.52it/s]
 22%|██▏       | 9/41 [00:04<00:15,  2.02it/s]
 29%|██▉       | 12/41 [00:04<00:09,  3.01it/s]
 41%|████▏     | 17/41 [00:05<00:05,  4.16it/s]
 49%|████▉     | 20/41 [00:05<00:04,  5.07it/s]
 56%|█████▌    | 23/41 [00:05<00:02,  6.72it/s]
 61%|██████    | 25/41 [00:06<00:02,  5.73it/s]
 68%|██████▊   | 28/41 [00:06<00:01,  7.35it/s]
 80%|████████  | 33/41 [00:06<00:00, 11.42it/s]
100%|██████████| 41/41 [00:06<00:00, 19.79it/s]
100%|██████████| 1/1 [00:07<00:00,  7.09s/it]Saving predictions to BBBP_preds.csv
Elapsed time = 0:00:07
```

```
bp_preds_df = pd.read_csv("BBBP_preds.csv")
bp_preds_df.head()
```

| | smiles | HIV_active |
|---|---|---|
| 0 | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 | 0.4572078287601471 |
| 1 | C(=O)(OC(C)(C)C)CCCc1ccc(cc1)N(CCCl)CCCl | 0.42620205879211426 |
| 2 | c12c3c(N4CCN(C)CC4)c(F)cc1c(c(C(O)=O)cn2C(C)CO... | 0.45636186003685 |
| 3 | C1CCN(CC1)Cc1cccc(c1)OCCCNC(=O)C | 0.4254920482635498 |
| 4 | Cc1onc(c2ccccc2Cl)c1C(=O)N[C@H]3[C@H]4SC(C)(C)... | 0.43473759293556213 |

Next steps: View recommended plots

```
bp_preds_df.tail()
```

| | smiles | HIV_active |
|---|---|---|
| 2045 | C1=C(Cl)C(=C(C2=C1NC(=O)C(N2)=O)[N+](=O)[O-])Cl | 0.4977163076400757 |
| 2046 | [C@H]3([N]2C1=C(C(=NC=N1)N)N=C2)[C@@H]([C@@H](... | 0.44215840101242065 |
| 2047 | [O+]1=N[N](C=C1[N-]C(NC2=CC=CC=C2)=O)C(CC3=CC=... | 0.465373158454895 |
| 2048 | C1=C(OC)C(=CC2=C1C(=[N+](C(=C2CC)C)[NH-])C3=CC... | 0.4850277900695801 |
| 2049 | [N+](=NCC(=O)N[C@@H]([C@H](O)C1=CC=C([N+]([O-]... | 0.44983288645744324 |

```
bp_preds_df.describe()
```

| | smiles | HIV_active |
|---|---|---|
| count | 2050 | 2050 |
| unique | 2050 | 1987 |
| top | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 | Invalid SMILES |
| freq | 1 | 11 |

```
bp_preds_df = bp_preds_df[bp_preds_df['HIV_active'] != "Invalid SMILES"]
bp_preds_df.describe()
```

| | smiles | HIV_active |
|---|---|---|
| count | 2039 | 2039 |
| unique | 2039 | 1986 |
| top | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 | 0.44356390833854675 |
| freq | 1 | 3 |

```
bp_preds_df['HIV_active'] = bp_preds_df['HIV_active'].astype(float)
```

```
bp_preds_df['HIV_active_2'] = bp_preds_df['HIV_active'].apply(lambda x: 1 if x > 0.4 else 0)
bp_preds_df.head()
```

1 to 5 of 5 entries Filter

| index | smiles | HIV_active | HIV_active_2 |
|---|---|---|---|
| 0 | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 | 0.4572078287601471 | 1 |
| 1 | C(=O)(OC(C)(C)C)CCCc1ccc(cc1)N(CCCl)CCCl | 0.42620205879211426 | 1 |
| 2 | c12c3c(N4CCN(C)CC4)c(F)cc1c(c(C(O)=O)cn2C(C)CO3)=O | 0.45636186003685 | 1 |
| 3 | C1CCN(CC1)Cc1cccc(c1)OCCCNC(=O)C | 0.4254920482635498 | 1 |
| 4 | Cc1onc(c2ccccc2Cl)c1C(=O)N[C@H]3[C@H]4SC(C)(C)[C@@H](N4C3=O)C(O)=O | 0.43473759293556213 | 1 |

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

Next steps: View recommended plots

```
bp_preds_df.describe()
```

| | HIV_active | HIV_active_2 |
|---|---|---|
| **count** | 2039.000000 | 2039.000000 |
| **mean** | 0.428393 | 0.780284 |
| **std** | 0.038881 | 0.414155 |
| **min** | 0.272080 | 0.000000 |
| **25%** | 0.404796 | 1.000000 |
| **50%** | 0.438196 | 1.000000 |
| **75%** | 0.456277 | 1.000000 |
| **max** | 0.509148 | 1.000000 |

```
# Filter rows where 'target_column' is equal to 1
bp_preds_df_filtered = bp_preds_df[bp_preds_df['HIV_active_2'] == 1]
bp_preds_df_filtered
```

| | smiles | HIV_active | HIV_active_2 |
|---|---|---|---|
| **0** | [Cl].CC(C)NCC(O)COc1cccc2ccccc12 | 0.457208 | 1 |
| **1** | C(=O)(OC(C)(C)C)CCCc1ccc(cc1)N(CCCl)CCCl | 0.426202 | 1 |
| **2** | c12c3c(N4CCN(C)CC4)c(F)cc1c(c(C(O)=O)cn2C(C)CO... | 0.456362 | 1 |
| **3** | C1CCN(CC1)Cc1cccc(c1)OCCCNC(=O)C | 0.425492 | 1 |
| **4** | Cc1onc(c2ccccc2Cl)c1C(=O)N[C@H]3[C@H]4SC(C)(C)... | 0.434738 | 1 |
| **...** | ... | ... | ... |
| **2045** | C1=C(Cl)C(=C(C2=C1NC(=O)C(N2)=O)[N+](=O)[O-])Cl | 0.497716 | 1 |
| **2046** | [C@H]3([N]2C1=C(C(=NC=N1)N)N=C2)[C@@H]([C@@H](... | 0.442158 | 1 |
| **2047** | [O+]1=N[N](C=C1[N-]C(NC2=CC=CC=C2)=O)C(CC3=CC=... | 0.465373 | 1 |
| **2048** | C1=C(OC)C(=CC2=C1C(=[N+](C(=C2CC)C)[NH-])C3=CC... | 0.485028 | 1 |

Next steps: View recommended plots

```
sub_df = pd.read_csv("substances.csv")
sub_df.head()
```

| | zinc_id | smiles |
|---|---|---|
| **0** | ZINC000000000027 | N[C@@H](CCc1ccc(N(CCCl)CCCl)cc1)C(=O)O |
| **1** | ZINC000016090786 | N[C@H](CCc1ccc(N(CCCl)CCCl)cc1)C(=O)O |
| **2** | ZINC000001763088 | N[C@H](CCCc1ccc(N(CCCl)CCCl)cc1)C(=O)O |
| **3** | ZINC000002033385 | N[C@@H](CCCc1ccc(N(CCCl)CCCl)cc1)C(=O)O |
| **4** | ZINC000000001673 | N[C@@H](Cc1ccc(N(CCCl)CCCl)cc1)C(=O)O |

Next steps: View recommended plots

```
sub_df.tail()
```

| | zinc_id | smiles |
|---|---|---|
| **46** | ZINC000196349655 | O=C(O)CCSc1ccc(N(CCCl)CCCl)cc1 |
| **47** | ZINC000064454242 | N=NCCCc1ccc(N(CCCl)CCCl)cc1 |
| **48** | ZINC000005161807 | O=C(O)C/C=C/c1ccc(N(CCCl)CCCl)cc1 |
| **49** | ZINC000001682294 | O=C(O)CCOc1ccc(N(CCCl)CCCl)cc1 |
| **50** | ZINC000079564304 | O=C(O)CNC(=O)c1ccc(N(CCCl)CCCl)cc1 |

```
sub_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51 entries, 0 to 50
Data columns (total 2 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   zinc_id  51 non-null     object
 1   smiles   51 non-null     object
dtypes: object(2)
memory usage: 944.0+ bytes
```

```
arguments = [
    '--test_path', 'substances.csv',
    '--preds_path', 'substances_preds.csv',
    '--checkpoint_dir', 'test_checkpoints_multimolecule',
    '--smiles_columns', 'smiles'
]

args = chemprop.args.PredictArgs().parse_args(arguments)
preds = chemprop.train.make_predictions(args=args)
```

```
Loading training args
Setting molecule featurization parameters to default.
Loading data
51it [00:00, 62002.75it/s]
100%|██████████| 51/51 [00:00<00:00, 85358.94it/s]
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: UserWarning: This DataLoader will cre
  warnings.warn(_create_warning_msg(
Validating SMILES
Test size = 51
  0%|          | 0/1 [00:00<?, ?it/s]Loading pretrained parameter "encoder.encoder.0.cached_zero_vector".
Loading pretrained parameter "encoder.encoder.0.W_i.weight".
Loading pretrained parameter "encoder.encoder.0.W_h.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.bias".
Loading pretrained parameter "readout.1.weight".
Loading pretrained parameter "readout.1.bias".
Loading pretrained parameter "readout.4.weight".
Loading pretrained parameter "readout.4.bias".
Moving model to cuda

  0%|          | 0/2 [00:00<?, ?it/s]
 50%|█████     | 1/2 [00:00<00:00,  2.70it/s]
100%|██████████| 1/1 [00:01<00:00,  1.17s/it]Saving predictions to substances_preds.csv
Elapsed time = 0:00:01
```

```
fda_df = pd.read_csv("fda_approved.csv")
fda_df.head()
```

| | zinc_id | smiles |
|---|---|---|
| 0 | ZINC000001530427 | C[C@@H]1O[C@@H]1P(=O)(O)O |
| 1 | ZINC000003807804 | Clc1ccccc1C(c1ccccc1)(c1ccccc1)n1ccnc1 |
| 2 | ZINC000000120286 | Nc1nc(N)c2nc(-c3ccccc3)c(N)nc2n1 |
| 3 | ZINC000242548690 | C[C@H]1O[C@@H](O[C@H]2[C@@H](O)C[C@H](O[C@H]3[... |
| 4 | ZINC000000008492 | Oc1cccc2cccnc12 |

Next steps: View recommended plots

```
arguments = [
    '--test_path', 'fda_approved.csv',
    '--preds_path', 'fda_approved_preds.csv',
    '--checkpoint_dir', 'test_checkpoints_multimolecule',
    '--smiles_columns', 'smiles'
]

args = chemprop.args.PredictArgs().parse_args(arguments)
preds = chemprop.train.make_predictions(args=args)
```

```
Loading training args
Setting molecule featurization parameters to default.
Loading data
892it [00:00, 193529.86it/s]
100%|██████████| 892/892 [00:00<00:00, 127442.15it/s]Validating SMILES
```

```
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: UserWarning: This DataLoader will cre
  warnings.warn(_create_warning_msg(
Test size = 892
  0%|          | 0/1 [00:00<?, ?it/s]Loading pretrained parameter "encoder.encoder.0.cached_zero_vector".
Loading pretrained parameter "encoder.encoder.0.W_i.weight".
Loading pretrained parameter "encoder.encoder.0.W_h.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.bias".
Loading pretrained parameter "readout.1.weight".
Loading pretrained parameter "readout.1.bias".
Loading pretrained parameter "readout.4.weight".
Loading pretrained parameter "readout.4.bias".
Moving model to cuda

  0%|          | 0/18 [00:00<?, ?it/s]
  6%|▌         | 1/18 [00:02<00:36,  2.14s/it]
 22%|██▏       | 4/18 [00:02<00:06,  2.25it/s]
 50%|█████     | 9/18 [00:02<00:01,  5.38it/s]
 94%|█████████▍| 17/18 [00:02<00:00, 11.85it/s]
100%|██████████| 1/1 [00:03<00:00,  3.04s/it]Saving predictions to fda_approved_preds.csv
Elapsed time = 0:00:04
```

```
fda_preds_df = pd.read_csv("fda_approved_preds.csv")
fda_preds_df.head()
```

| | zinc_id | smiles | HIV_active |
|---|---|---|---|
| 0 | ZINC000001530427 | C[C@@H]1O[C@@H]1P(=O)(O)O | 0.381745 |
| 1 | ZINC000003807804 | Clc1ccccc1C(c1ccccc1)(c1ccccc1)n1ccnc1 | 0.477443 |
| 2 | ZINC000000120286 | Nc1nc(N)c2nc(-c3ccccc3)c(N)nc2n1 | 0.496834 |
| 3 | ZINC000242548690 | C[C@H]1O[C@@H](O[C@H]2[C@@H](O)C[C@H](O[C@H]3[... | 0.347241 |
| 4 | ZINC000000008492 | Oc1cccc2cccnc12 | 0.487933 |

Next steps: View recommended plots

```
fda_preds_df = fda_preds_df[fda_preds_df['HIV_active'] != "Invalid SMILES"]
fda_preds_df.describe()
fda_preds_df['HIV_active'] = fda_preds_df['HIV_active'].astype(float)
fda_preds_df['HIV_active_2'] = fda_preds_df['HIV_active'].apply(lambda x: 1 if x > 0.45 else 0)
fda_preds_df.head()
```

| | HIV_active |
|---|---|
| count | 892.000000 |
| mean | 0.437195 |
| std | 0.038624 |
| min | 0.303106 |
| 25% | 0.418839 |
| 50% | 0.446813 |
| 75% | 0.463876 |
| max | 0.510538 |

| | zinc_id | smiles | HIV_active | HIV_active_2 |
|---|---|---|---|---|
| 0 | ZINC000001530427 | C[C@@H]1O[C@@H]1P(=O)(O)O | 0.381745 | 0 |
| 1 | ZINC000003807804 | Clc1ccccc1C(c1ccccc1)(c1ccccc1)n1ccnc1 | 0.477443 | 1 |
| 2 | ZINC000000120286 | Nc1nc(N)c2nc(-c3ccccc3)c(N)nc2n1 | 0.496834 | 1 |
| | | C[C@H]1O[C@@H] | | |

Next steps: View recommended plots View recommended plots

```
# Filter rows where 'target_column' is equal to 1
fda_preds_df_filtered = fda_preds_df[fda_preds_df['HIV_active_2'] == 1]
fda_preds_df_filtered
```

| | zinc_id | smiles | HIV_active |
|---|---|---|---|
| **1** | ZINC000003807804 | Clc1ccccc1C(c1ccccc1)(c1ccccc1)n1ccnc1 | 0.477443 |
| **2** | ZINC000000120286 | Nc1nc(N)c2nc(-c3ccccc3)c(N)nc2n1 | 0.496834 |
| **4** | ZINC000000008492 | Oc1cccc2cccnc12 | 0.487933 |
| **5** | ZINC000003607120 | COc1c(N2CCN[C@H](C)C2)c(F)cc2c(=O)c(C(=O)O)cn(... | 0.457250 |
| **8** | ZINC000051133897 | CN1C(C(=O)Nc2ccccn2)=C(O)c2ccccc2S1(=O)=O | 0.467989 |
| **...** | ... | ... | ... |
| **878** | ZINC000003776633 | Cc1ccc(/C(=C\CN2CCCC2)c2cccc(/C=C/C(=O)O)n2)cc1 | 0.453140 |
| **882** | ZINC000003782818 | CCOc1nc2cccc(C(=O)O)c2n1Cc1ccc(-c2ccccc2-c2nnn... | 0.489798 |
| **883** | ZINC000003816292 | COc1cc2nccc(Oc3ccc(NC(=O)NC4CC4)c(Cl)c3)c2cc1C... | 0.475386 |
| **887** | ZINC000000537964 | O[C@H](c1cc(C(F)(F)F)nc2c(C(F)(F)F)cccc12)[C@H... | 0.455411 |
| **890** | ZINC000034636383 | COc1ccc(CC(C)(C)NC[C@H](O)c2cc(O)cc3c2OCC(=O)N... | 0.453037 |

408 rows × 4 columns

Next steps: View recommended plots

```
!wget https://zinc15.docking.org/substances/subsets/named.csv
```

```
--2024-03-10 05:23:09--  https://zinc15.docking.org/substances/subsets/named.csv
Resolving zinc15.docking.org (zinc15.docking.org)... 169.230.75.4
Connecting to zinc15.docking.org (zinc15.docking.org)|169.230.75.4|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/csv]
Saving to: ‘named.csv’

named.csv               [ <=>                ]   9.28K  --.-KB/s    in 0.04s

2024-03-10 05:23:10 (242 KB/s) - ‘named.csv’ saved [9499]
```

```
zinc_df = pd.read_csv("named.csv")
zinc_df.head()
```

| | zinc_id | smiles |
|---|---|---|
| **0** | ZINC000030727788 | C=C[C@]1(C)C[C@@H](OC(=O)CSC(C)(C)CNC(=O)[C@H]... |
| **1** | ZINC000150377216 | CCCCCC/C=C\C/C=C\CCCCCCCC(=O)OC[C@H](COCCCCCCC... |
| **2** | ZINC000100780125 | CC(=O)O[C@H]1C[C@](C)(O)[C@@H]2CC=C(C)[C@@H]2[... |
| **3** | ZINC000006580536 | O=C(O)[C@H](Cc1ccccc1)N(CCCl)CCCl |
| **4** | ZINC000150351802 | O=C1C[C@H](c2ccc(O)c(O)c2)Oc2c1c(O)cc(O[C@H]1O... |

Next steps: View recommended plots

```
arguments = [
    '--test_path', 'named.csv',
    '--preds_path', 'named_preds.csv',
    '--checkpoint_dir', 'test_checkpoints_multimolecule',
    '--smiles_columns', 'smiles'
]

args = chemprop.args.PredictArgs().parse_args(arguments)
preds = chemprop.train.make_predictions(args=args)
```

```
Loading training args
Setting molecule featurization parameters to default.
Loading data
100it [00:00, 59764.95it/s]
100%|██████████| 100/100 [00:00<00:00, 66905.47it/s]
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:557: UserWarning: This DataLoader will cre
  warnings.warn(_create_warning_msg(
Validating SMILES
Test size = 100
  0%|          | 0/1 [00:00<?, ?it/s]Loading pretrained parameter "encoder.encoder.0.cached_zero_vector".
Loading pretrained parameter "encoder.encoder.0.W_i.weight".
Loading pretrained parameter "encoder.encoder.0.W_h.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.weight".
Loading pretrained parameter "encoder.encoder.0.W_o.bias".
Loading pretrained parameter "readout.1.weight".
Loading pretrained parameter "readout.1.bias".
Loading pretrained parameter "readout.4.weight".
Loading pretrained parameter "readout.4.bias".
Moving model to cuda

  0%|          | 0/2 [00:00<?, ?it/s]
 50%|█████     | 1/2 [00:00<00:00,  1.59it/s]
100%|██████████| 1/1 [00:00<00:00,  1.01it/s]Saving predictions to named_preds.csv
Elapsed time = 0:00:01
```

```
zinc_preds_df = pd.read_csv("named_preds.csv")
zinc_preds_df.head()
zinc_preds_df = zinc_preds_df[zinc_preds_df['HIV_active'] != "Invalid SMILES"]
zinc_preds_df.describe()
zinc_preds_df['HIV_active'] = zinc_preds_df['HIV_active'].astype(float)
zinc_preds_df['HIV_active_2'] = zinc_preds_df['HIV_active'].apply(lambda x: 1 if x > 0.5 else 0)
zinc_preds_df.head()
```

| | zinc_id | smiles | HIV_active |
|---|---|---|---|
| 0 | ZINC000030727788 | C=C[C@]1(C)C[C@@H](OC(=O)CSC(C)(C)CNC(=O)[C@H]... | 0.359023 |
| 1 | ZINC000150377216 | CCCCCC/C=C\C/C=C\CCCCCCCC(=O)OC[C@H](COCCCCCCC... | 0.395612 |
| 2 | ZINC000100780125 | CC(=O)O[C@H]1C[C@](C)(O)[C@@H]2CC=C(C)[C@@H]2[... | 0.376613 |
| 3 | ZINC000006580536 | O=C(O)[C@H](Cc1ccccc1)N(CCCl)CCCl | 0.419707 |
| 4 | ZINC000150351802 | O=C1C[C@H](c2ccc(O)c(O)c2)Oc2c1c(O)cc(O[C@H]1O... | 0.459242 |

| | HIV_active |
|---|---|
| count | 100.000000 |
| mean | 0.410988 |
| std | 0.036220 |
| min | 0.312791 |
| 25% | 0.392779 |
| 50% | 0.407688 |
| 75% | 0.434701 |
| max | 0.503582 |

| | zinc_id | smiles | HIV_active | HIV_ac |
|---|---|---|---|---|
| 0 | ZINC000030727788 | C=C[C@]1(C)C[C@@H](OC(=O)CSC(C)(C)CNC(=O)[C@H]... | 0.359023 | |
| 1 | ZINC000150377216 | CCCCCC/C=C\C/C=C\CCCCCCCC(=O)OC[C@H](COCCCCCCC... | 0.395612 | |

Next steps: View recommended plots View recommended plots View recommended plots

```
# Filter rows where 'target_column' is equal to 1
zinc_preds_df_filtered = zinc_preds_df[zinc_preds_df['HIV_active_2'] == 1]
zinc_preds_df_filtered
```

| | zinc_id | smiles | HIV_active | HIV_active_2 |
|---|---|---|---|---|
| 72 | ZINC000001680645 | Nc1cccc2cc(S(=O)(=O)O)ccc12 | 0.503582 | 1 |

```
from google.colab import drive
drive.mount('/content/drive')
```