Entrenamientos Cierre

Después de variar el target en los entrenamientos de desarrollo encontramos que se obtienen los mejores resultados cuando el target es la energía total menos el offset. Por lo que con base en eso, ahora añadiremos como co-target a las fuerzas y modificaremos el nivel de importancia en el entrenamiento.

## Targets en el costo total

En el código el costo total está definido de la siguiente manera:

*# total loss*

*loss = loss\_y \* self.hparams.y\_weight + loss\_neg\_dy \* self.hparams.neg\_dy\_weight*

Donde loss\_y es el MSE del target “y”, en este caso es la energía total menos el offset;

self.hparams.y\_weight es un hiperparámetro que indica el nivel de peso de “y” en la función de costo total;

loss\_neg\_dy es el MSE del target neg\_dy, en este caso es la derivada negativa de la energía respecto a la posición que es lo mismo que la fuerza;

hparams.neg\_dy\_weight es un hiperparámetro que indica nivel de peso de “neg\_dy” en la función costo total.

En análisis de los entrenamientos es completamente análogo al análisis del entrenamiento con target Energía total menos el offset.

Los entrenamientos se realizaron con 4 GPU’s Nvidia A100

A continuación se muestran los resultados de los entrenamientos.

# y\_weight:1.0

La duración del entrenamiento es de 12 horas, 31 minutos y 59 segundos.

Configuración relevante:

batch\_size: 32

cutoff\_upper: 6.35

derivative: True

y\_weight: 1.0

neg\_dy\_weight: 1.0

inference\_batch\_size: 4

max\_num\_neighbors: 140

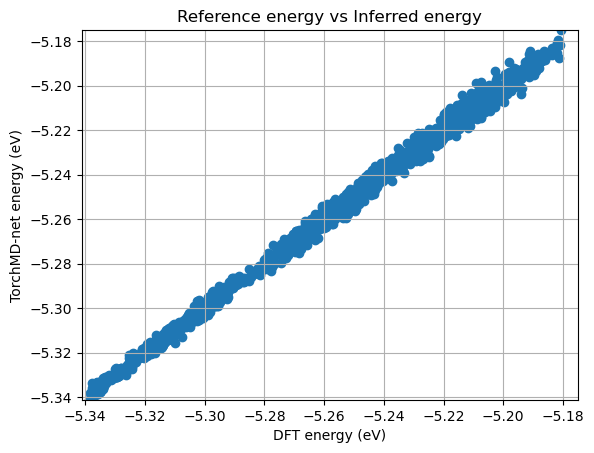
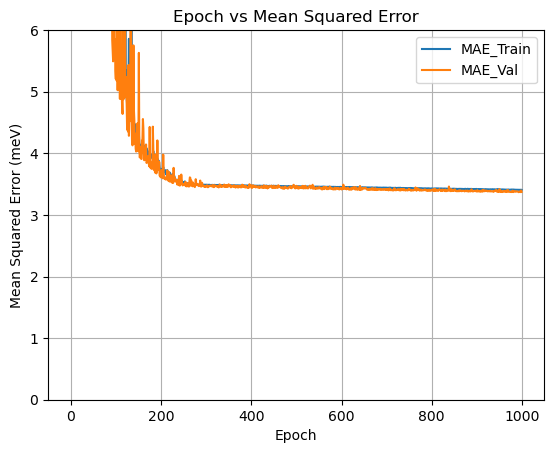
num\_epochs: 1000

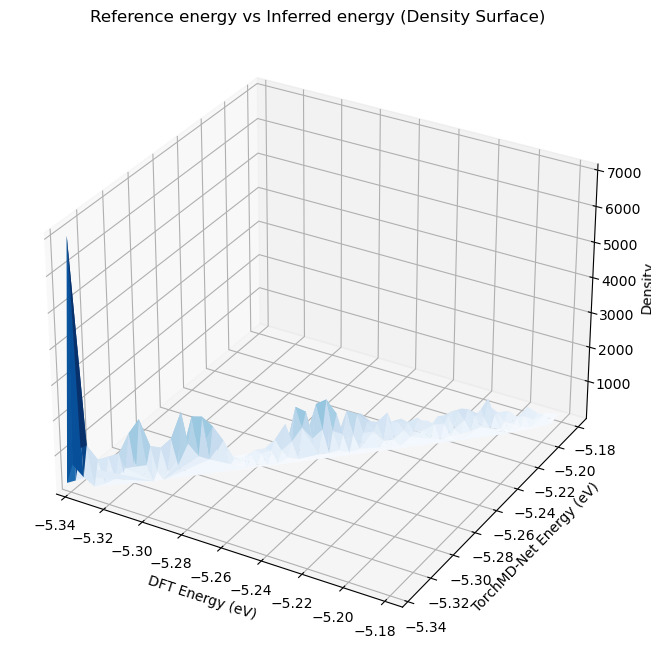
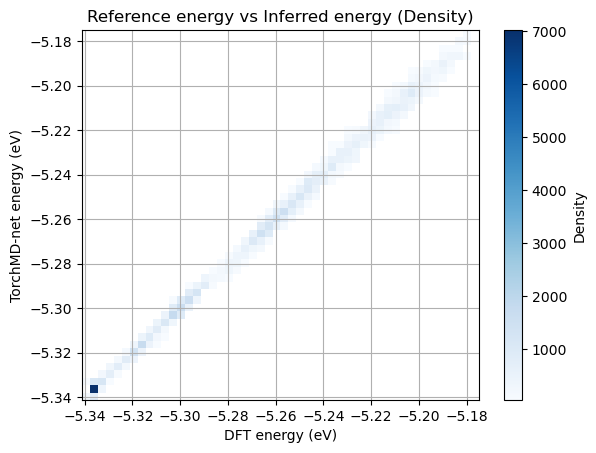
seed: 42

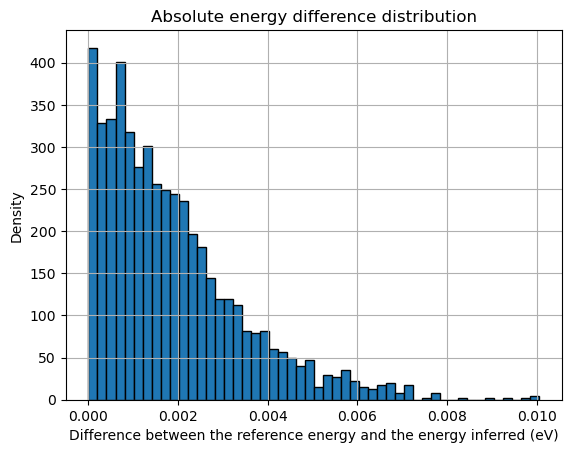
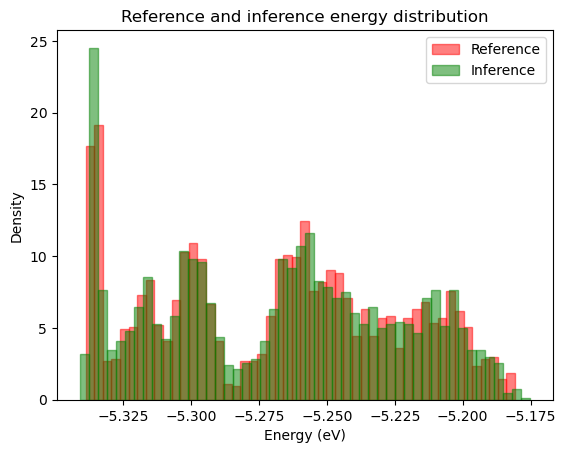
test\_size: 0.1

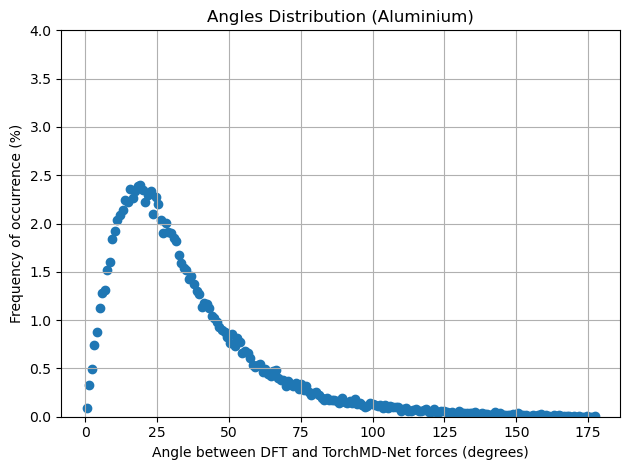
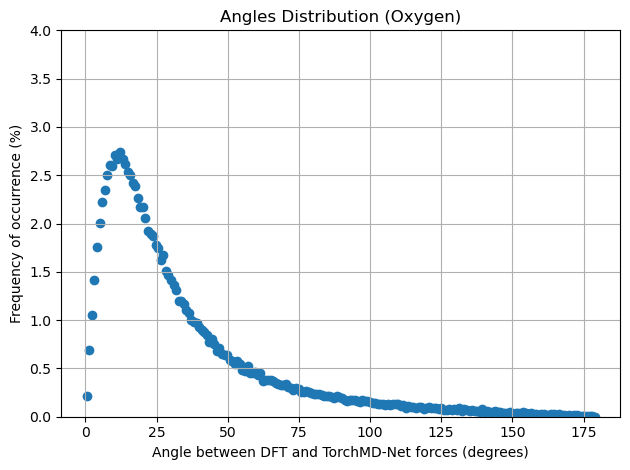
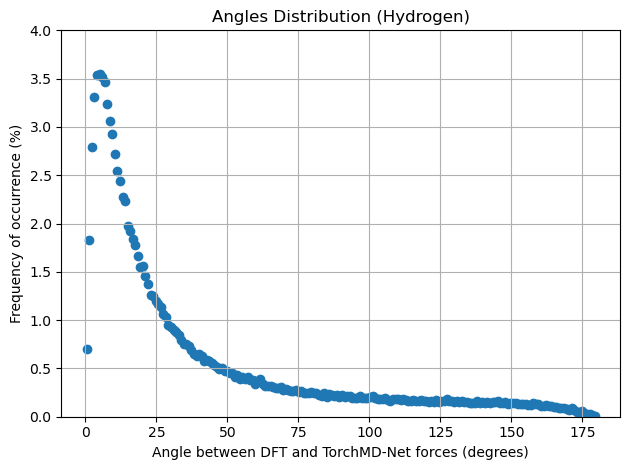
train\_size: 0.81

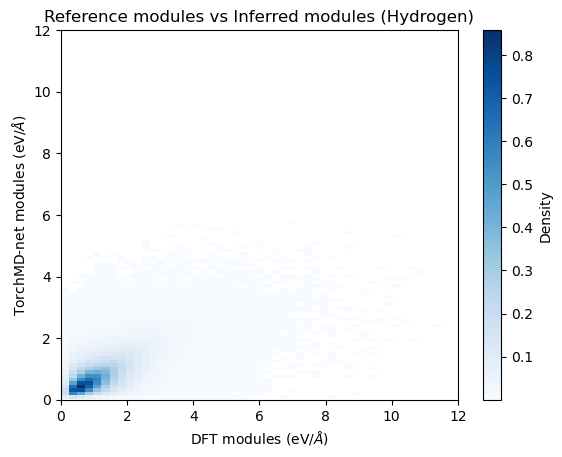
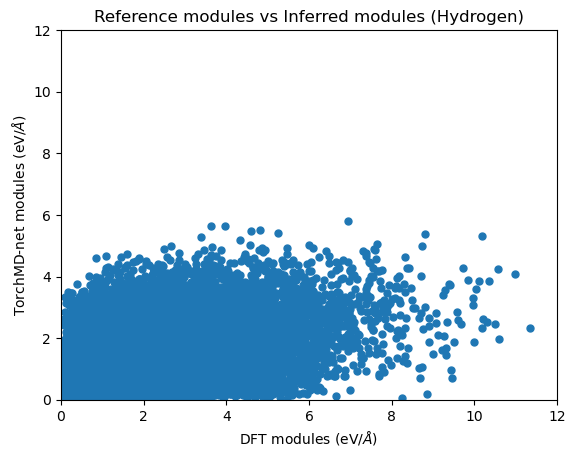
val\_size: 0.09

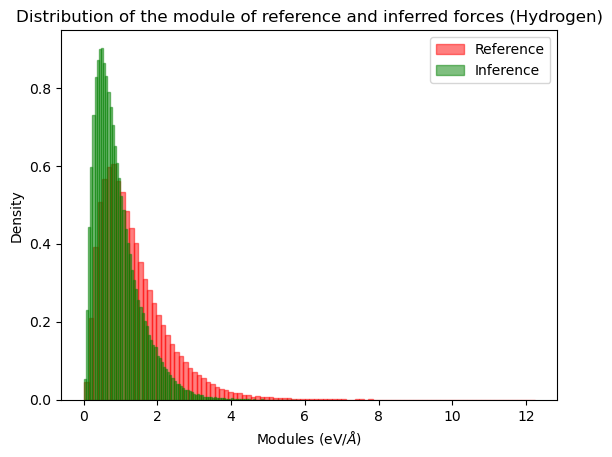
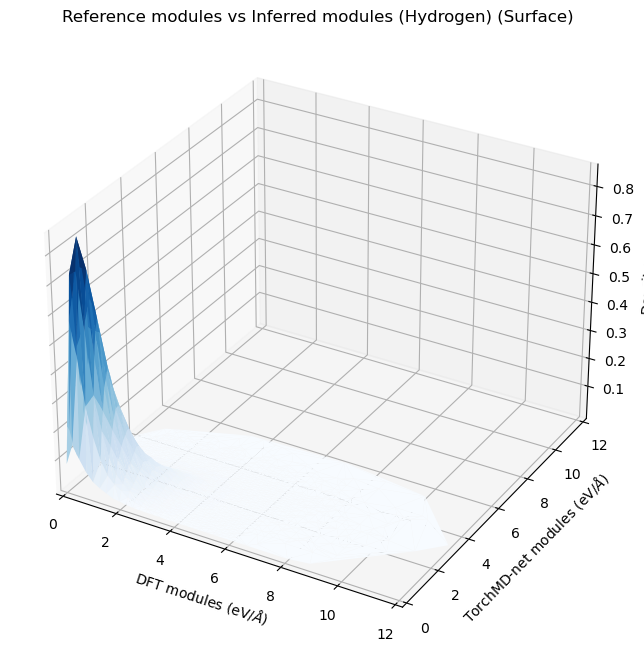


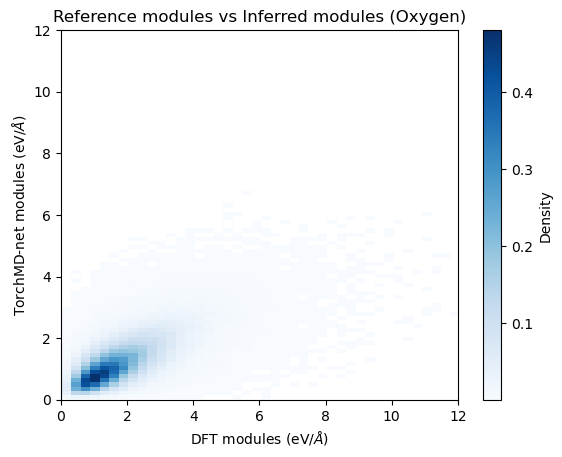
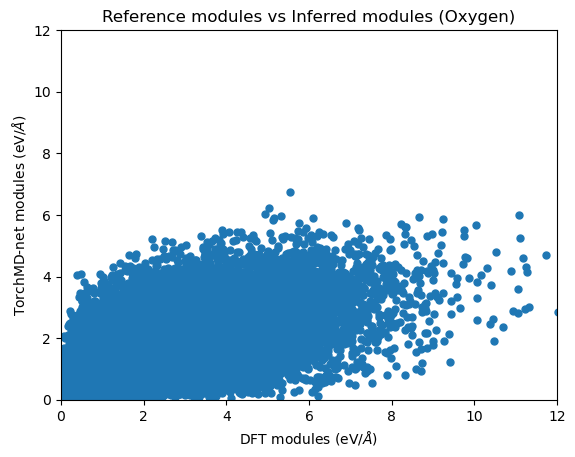


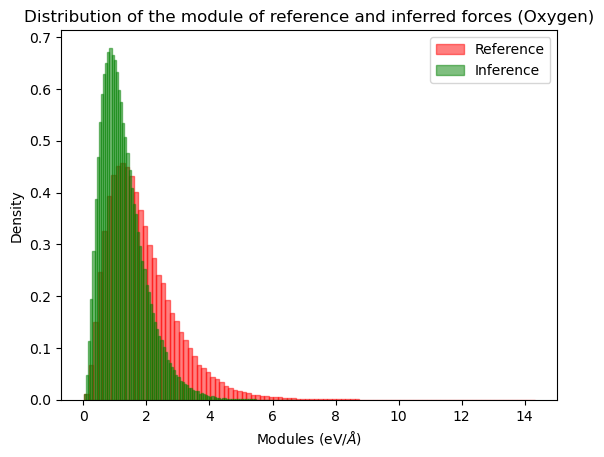
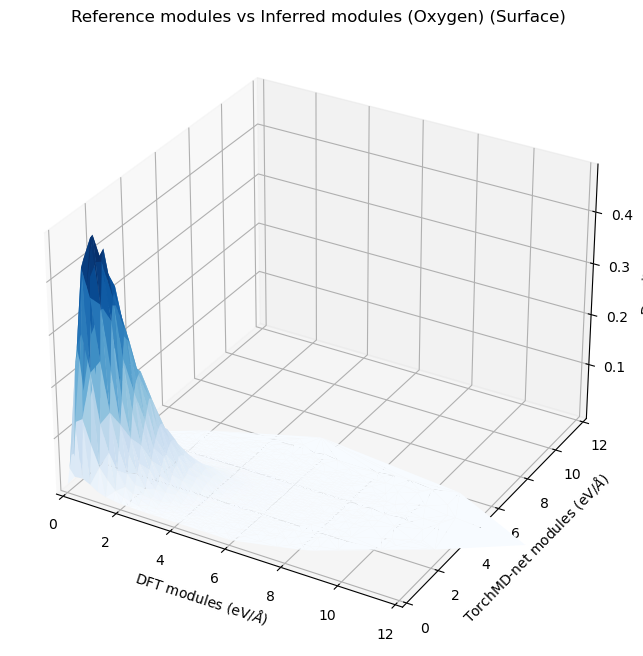


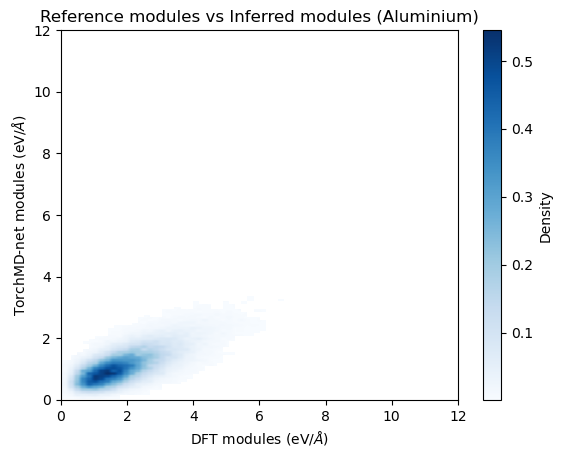
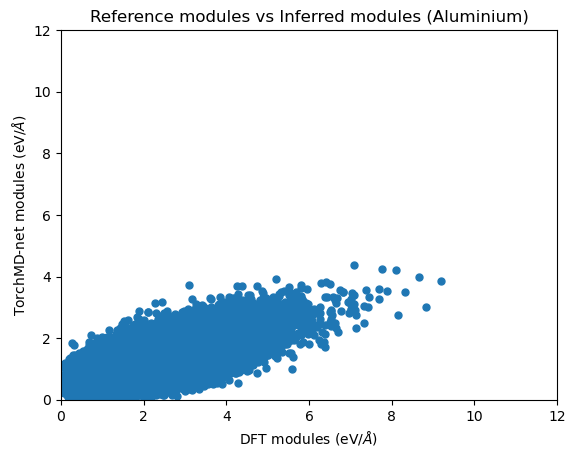


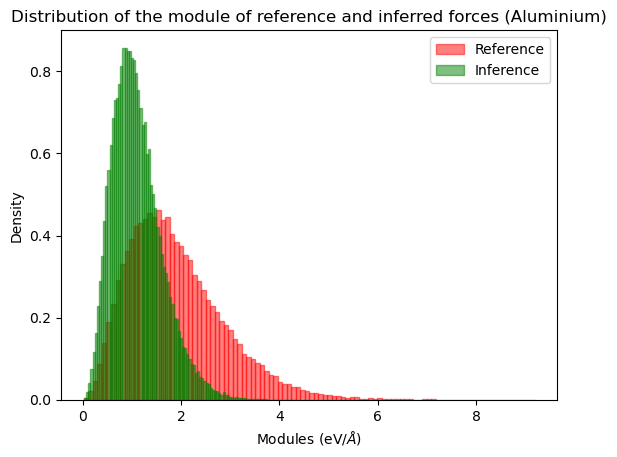
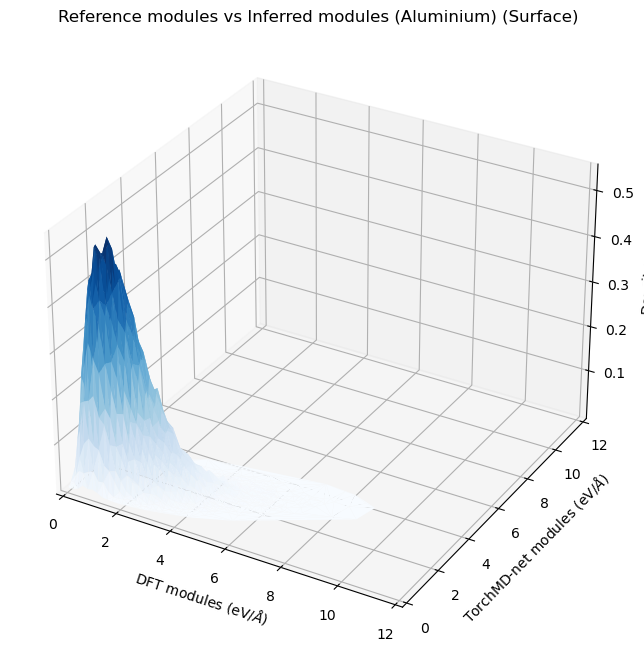












# y\_weight: 0.05

Este entrenamiento tiene como target la energía total menos la atómica.

La duración del entrenamiento es de 12 horas, 40 minutos y 35 segundos.

Configuración relevante:

batch\_size: 32

cutoff\_upper: 6.35

derivative: True

y\_weight: 0.05

neg\_dy\_weight: 1.0

inference\_batch\_size: 4

max\_num\_neighbors: 140

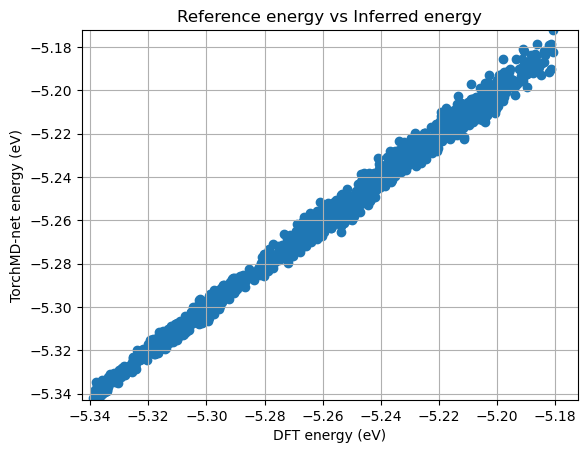
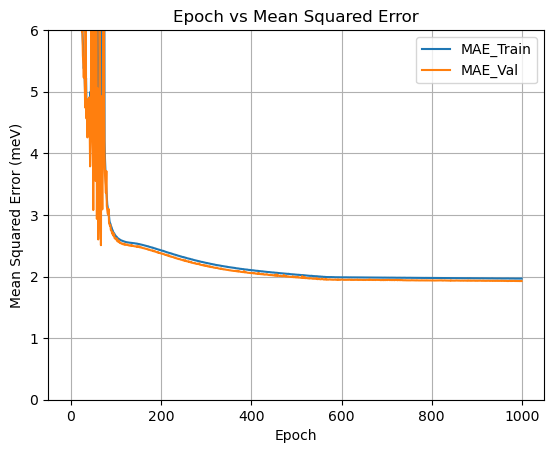
num\_epochs: 1000

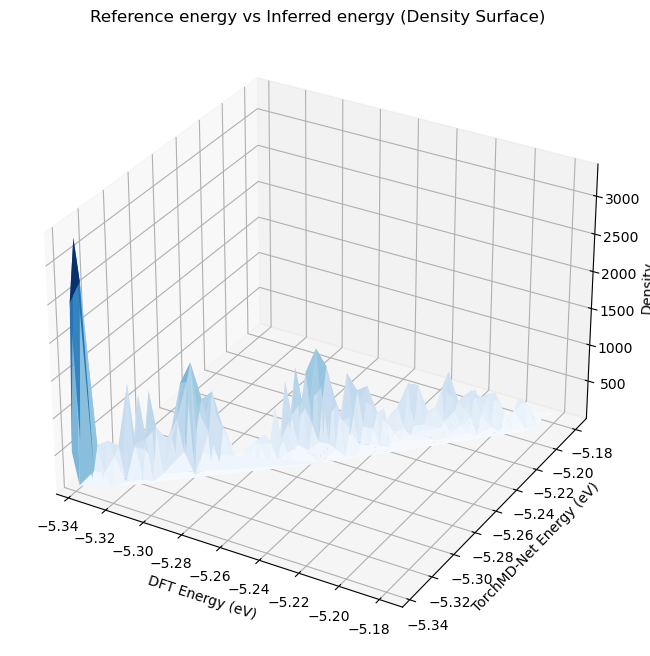
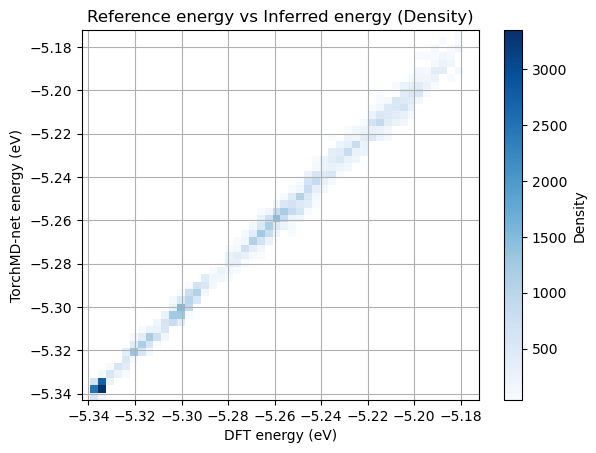
seed: 42

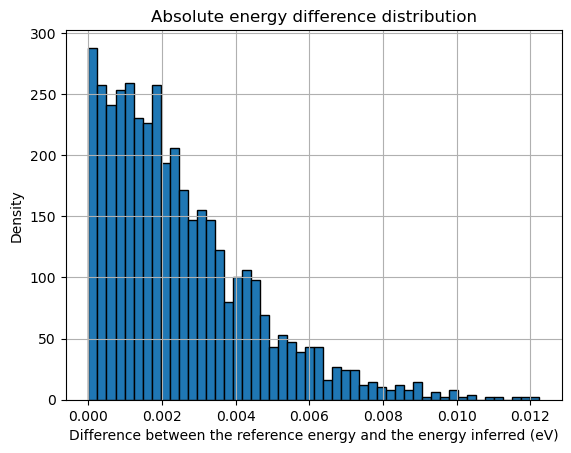
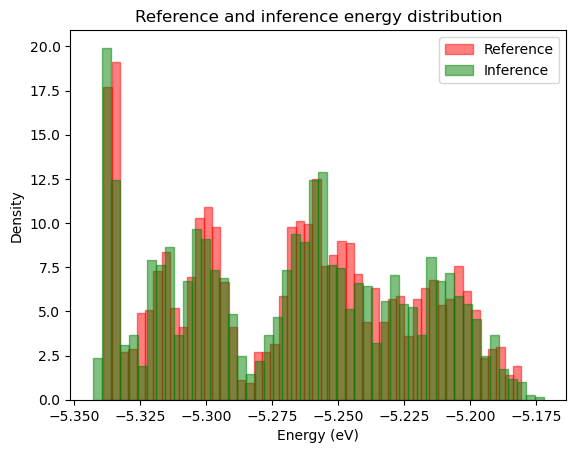
test\_size: 0.1

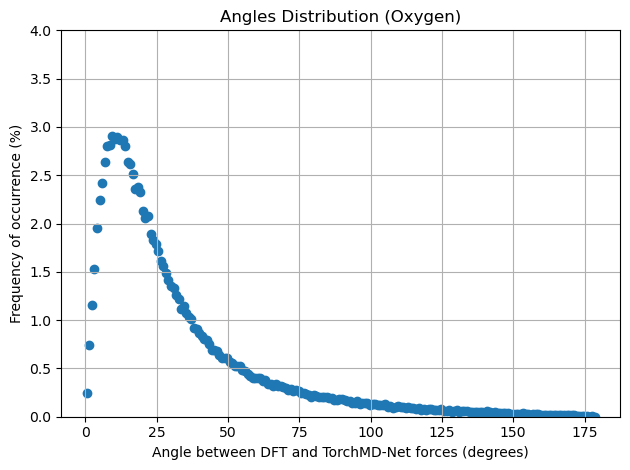
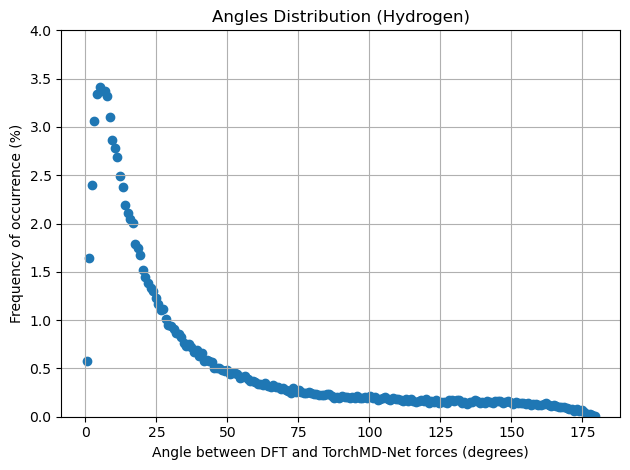
train\_size: 0.81

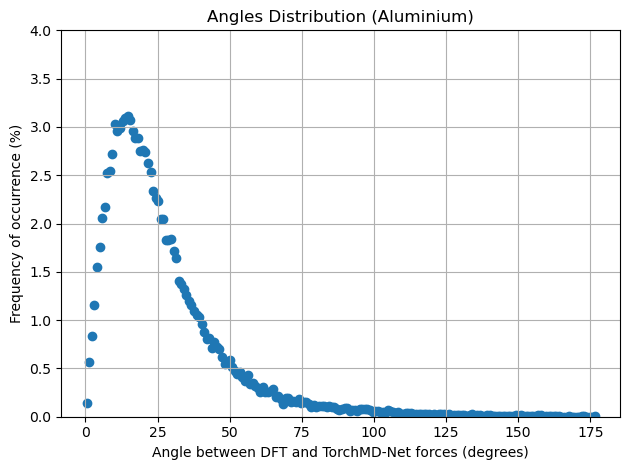
val\_size: 0.09

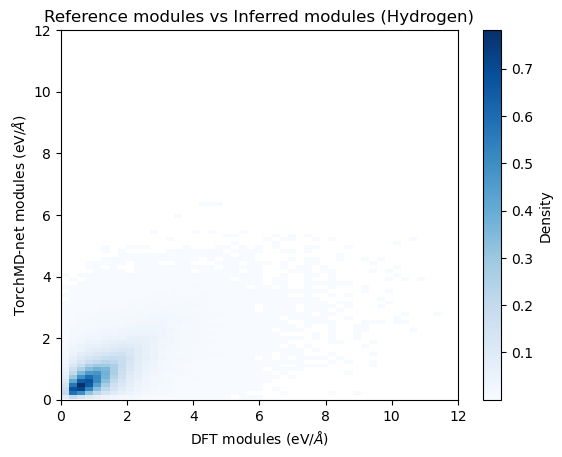
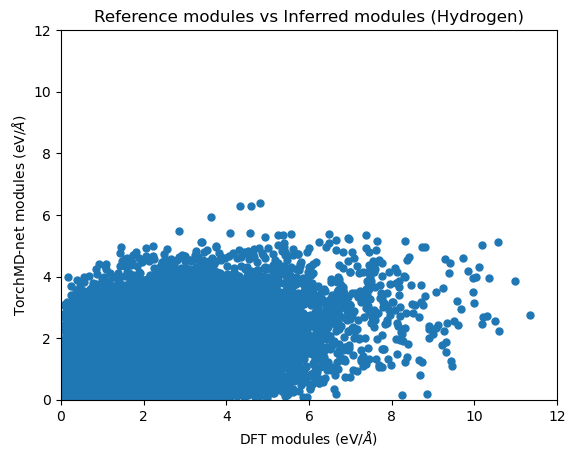


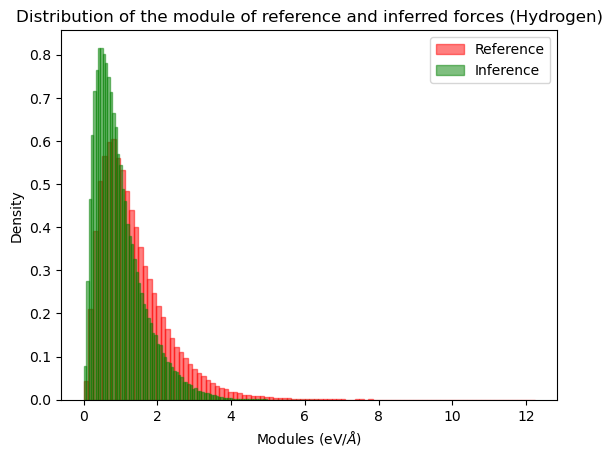
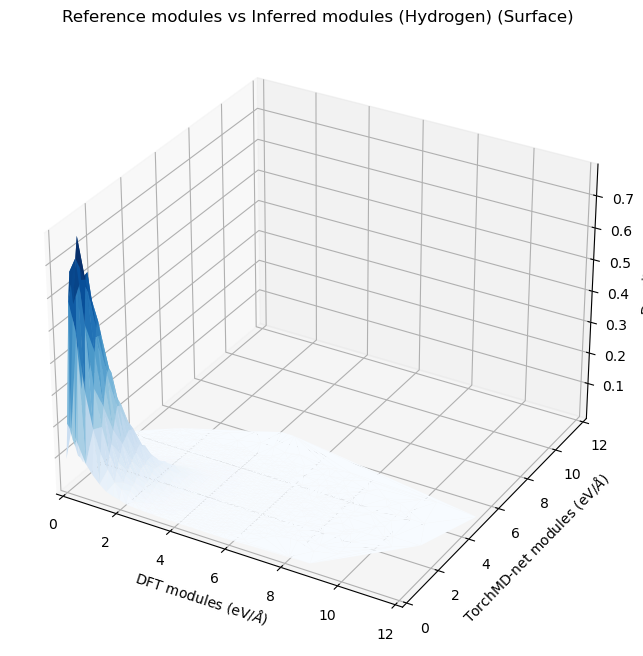


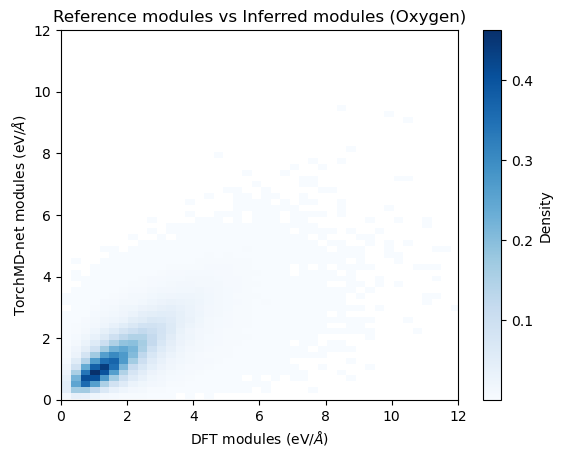
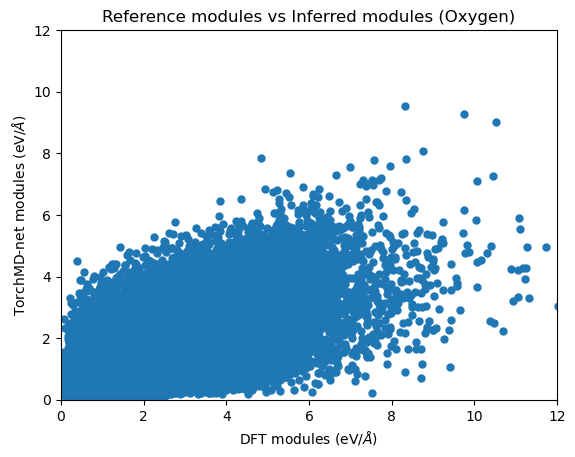


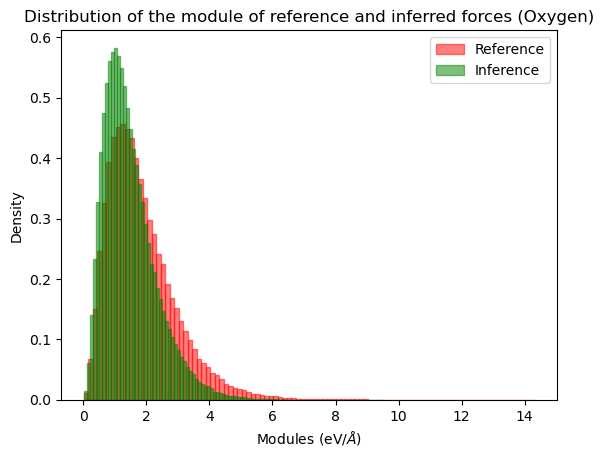
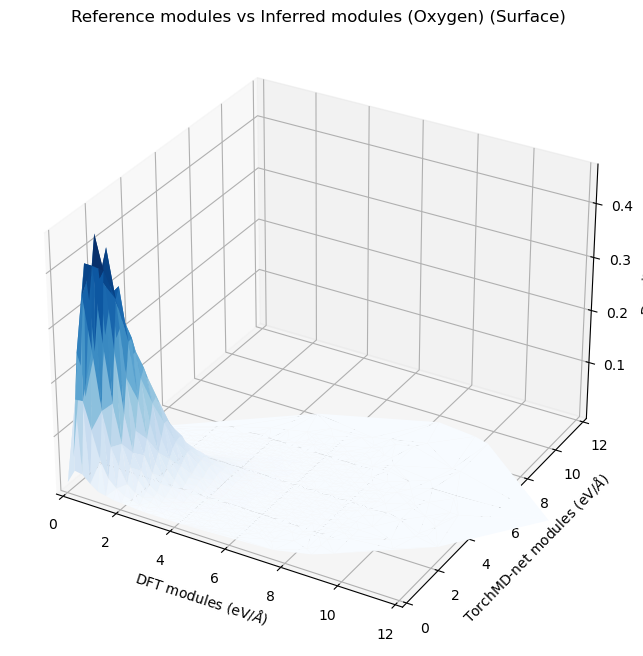


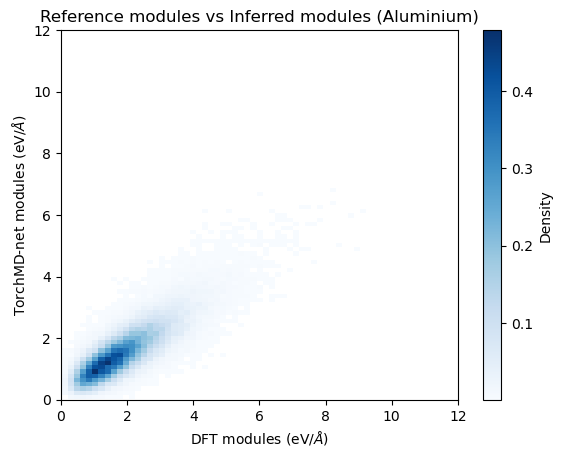
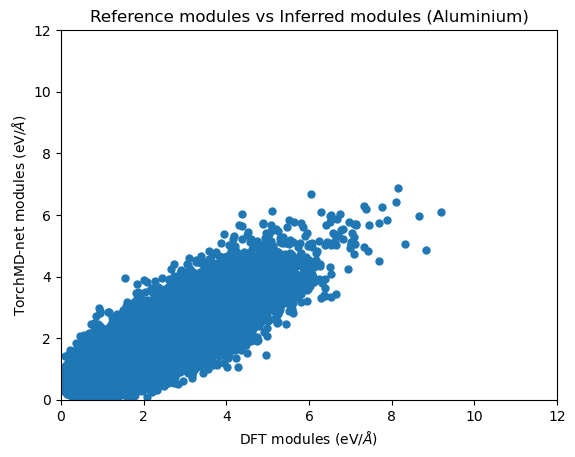


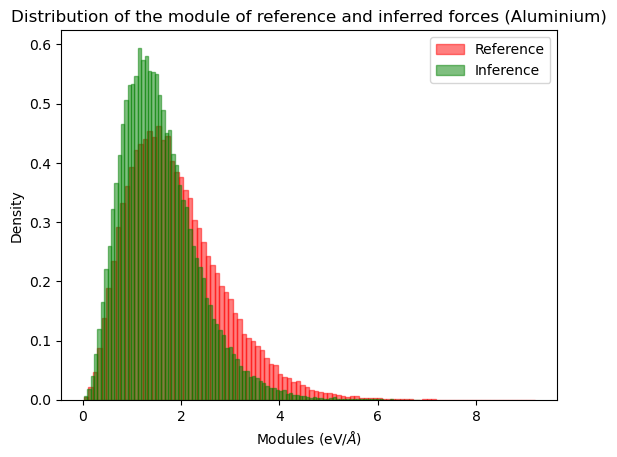
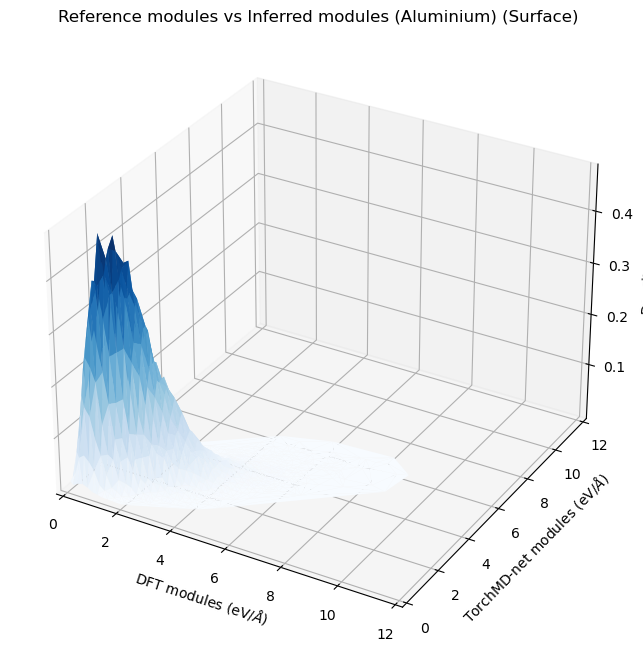












# y\_weight: 0.01

La duración del entrenamiento es de 7 horas, 12 minutos y 36 segundos.

Configuración relevante:

batch\_size: 32

cutoff\_upper: 6.35

derivative: True

y\_weight: 0.01

neg\_dy\_weight: 1.0

inference\_batch\_size: 4

max\_num\_neighbors: 140

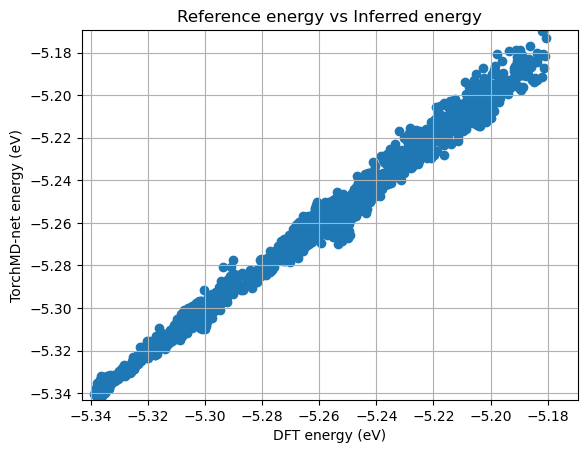
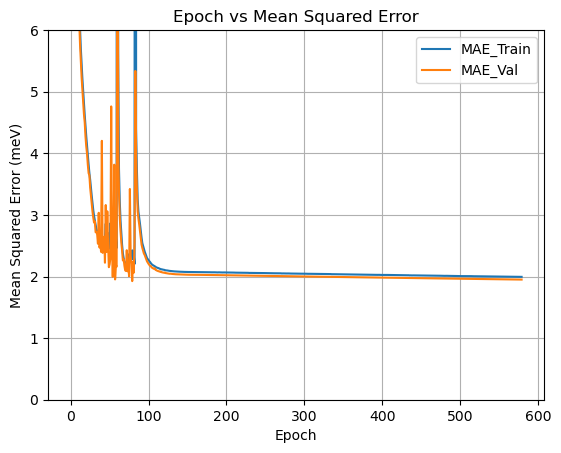
num\_epochs: 1000

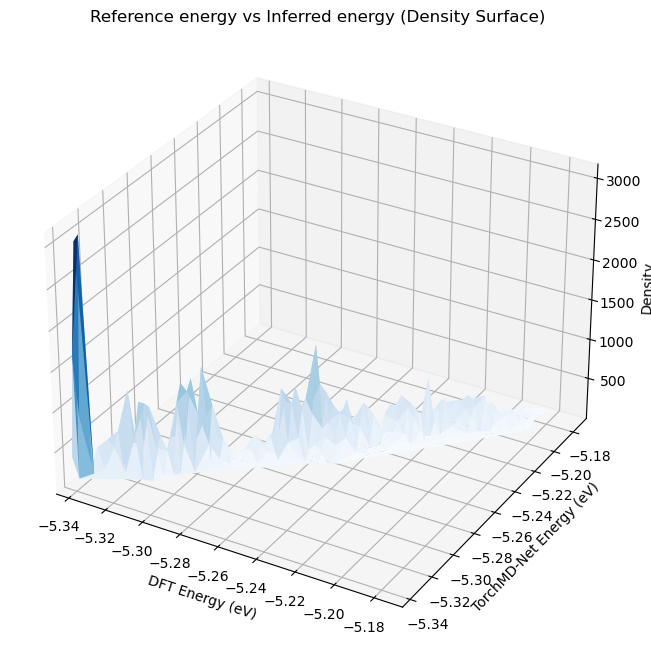
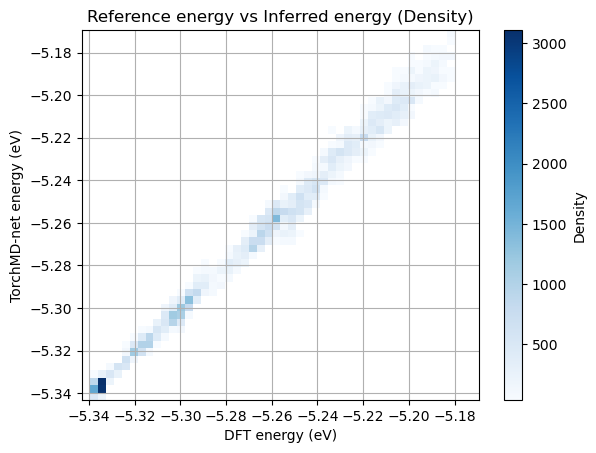
seed: 42

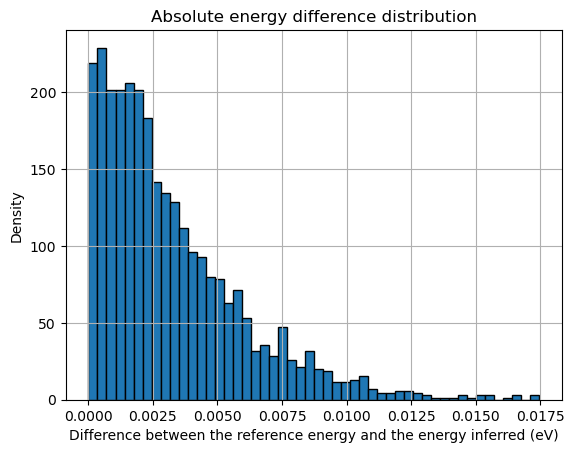
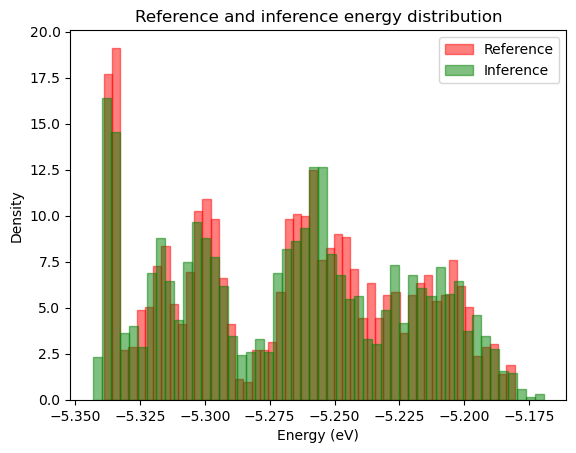
test\_size: 0.1

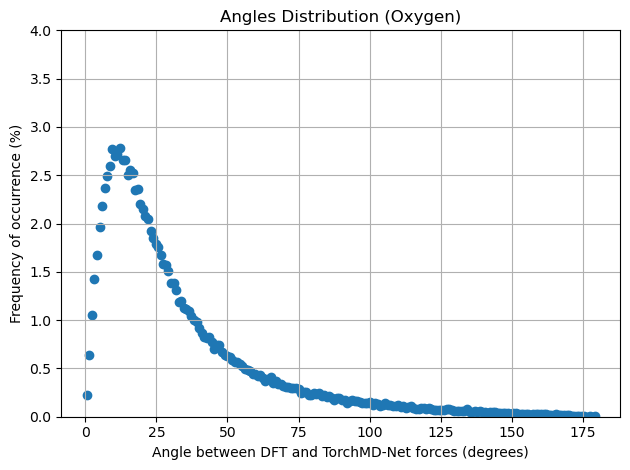
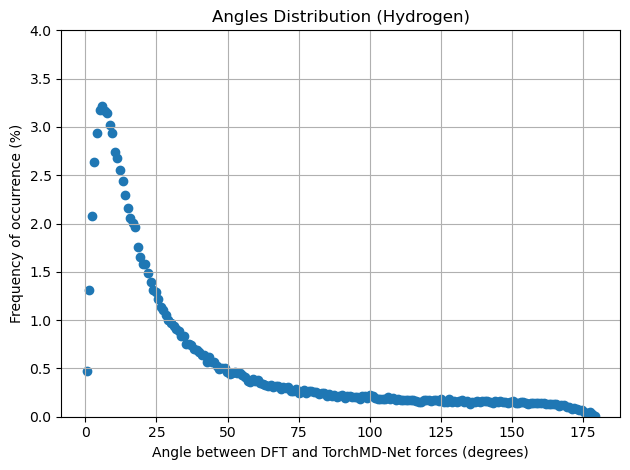
train\_size: 0.81

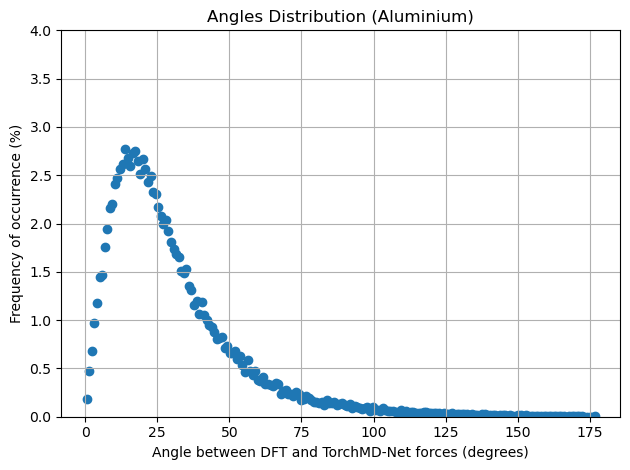
val\_size: 0.09

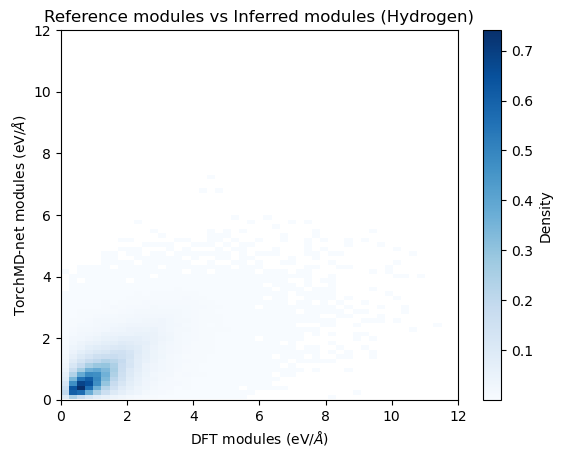
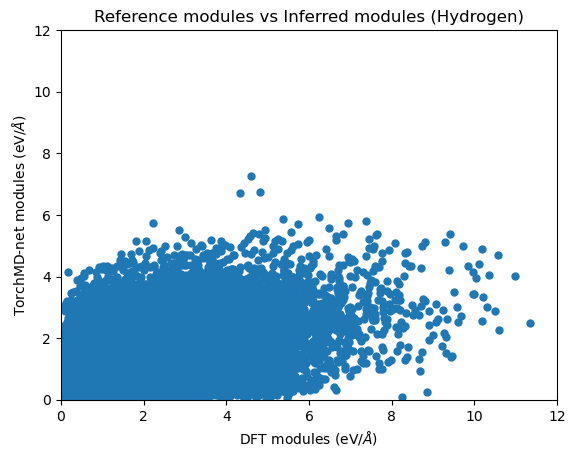


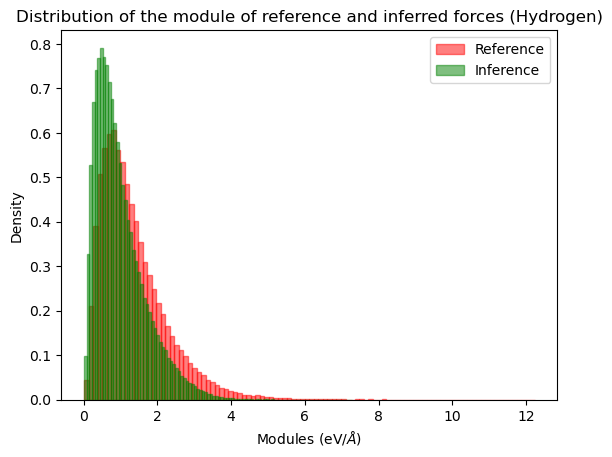
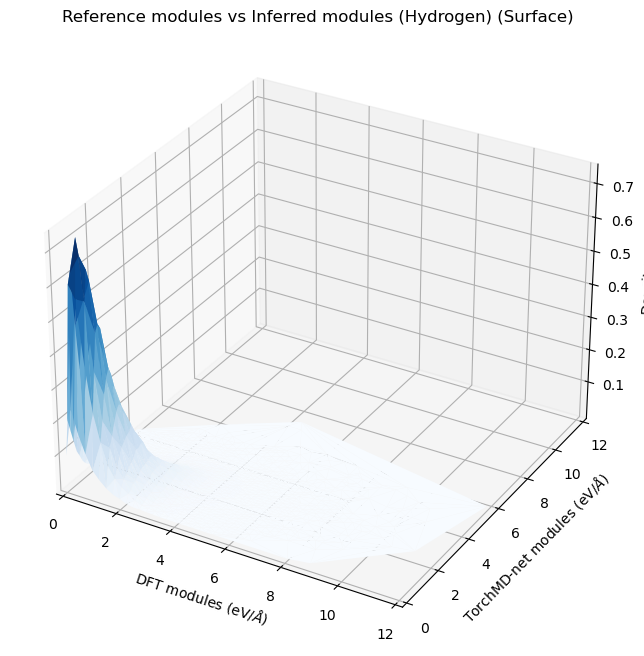


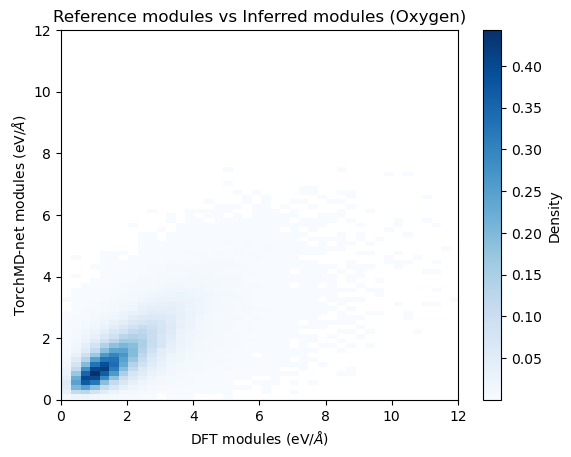
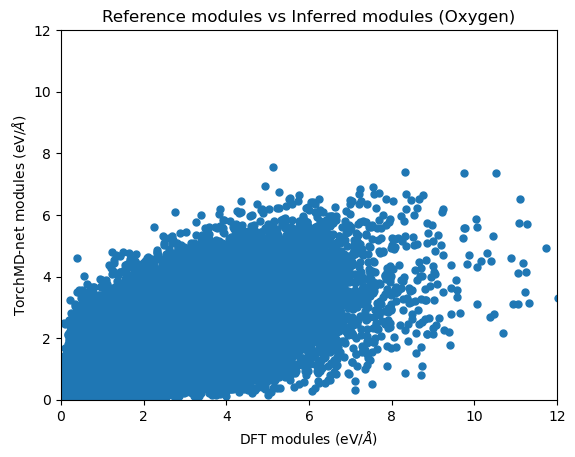


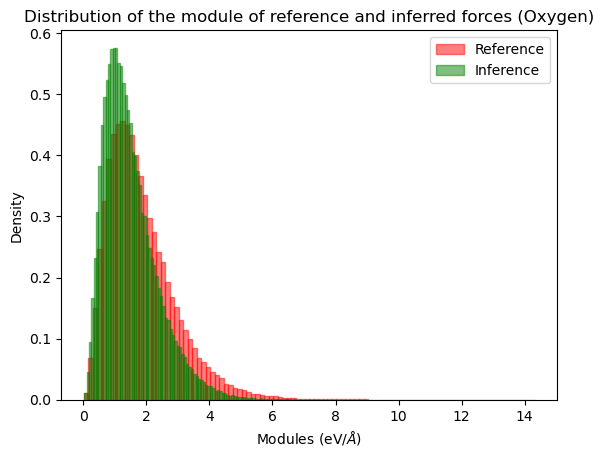
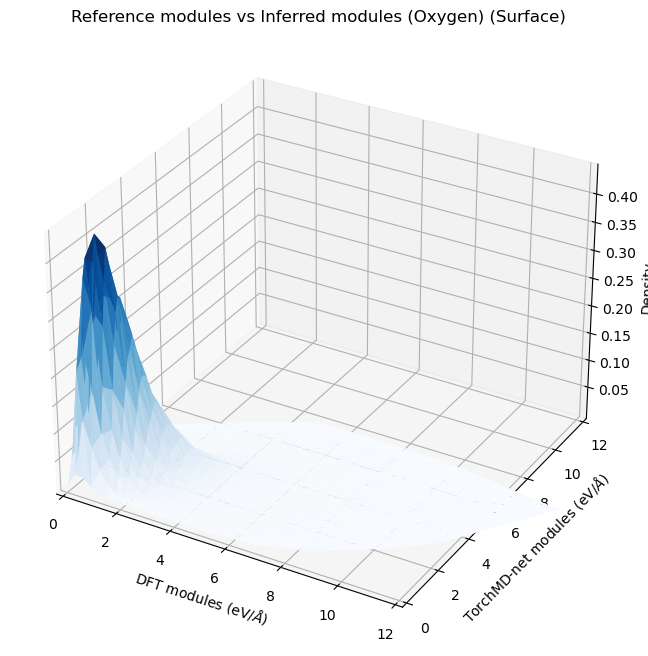


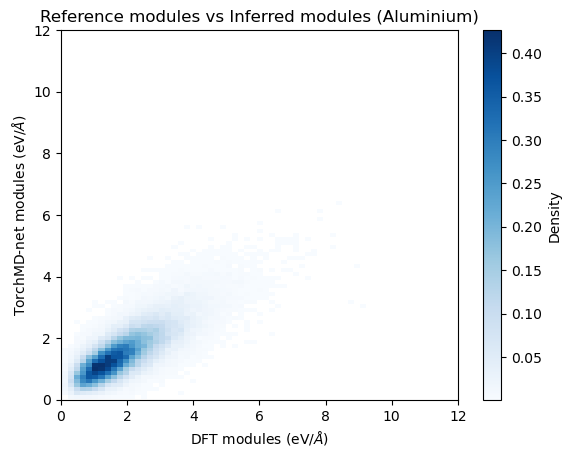
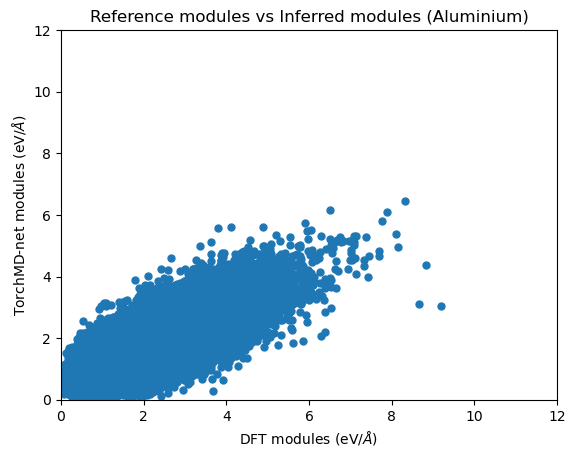


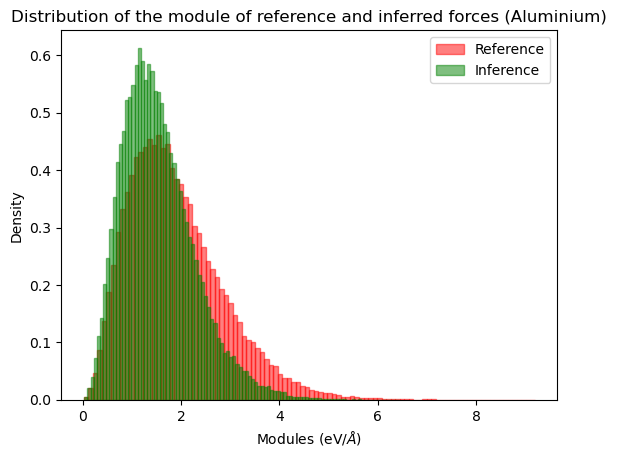
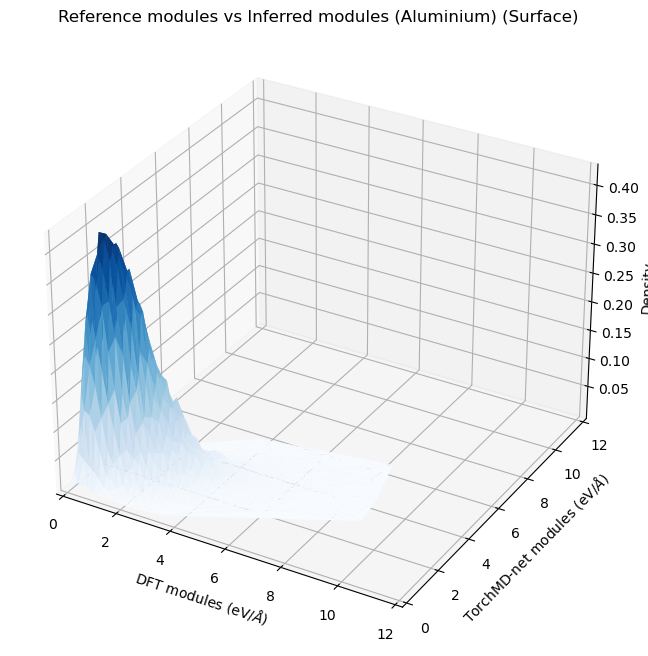












# y\_weight: 0.0001

La duración del entrenamiento es de 7 horas, 5 minutos y 28 segundos.

Configuración relevante:

batch\_size: 32

cutoff\_upper: 6.35

derivative: True

y\_weight: 0.0001

neg\_dy\_weight: 1.0

inference\_batch\_size: 4

max\_num\_neighbors: 140

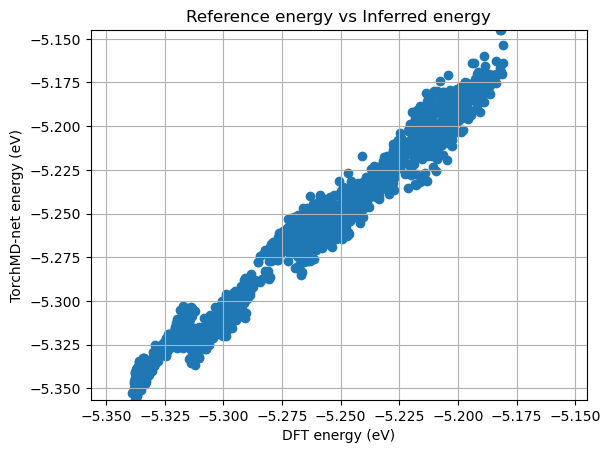
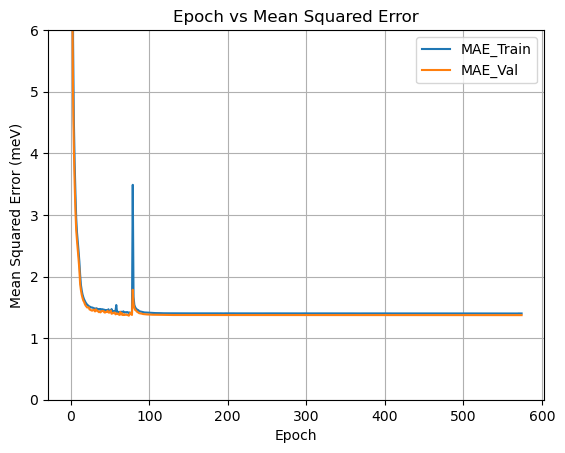
num\_epochs: 1000

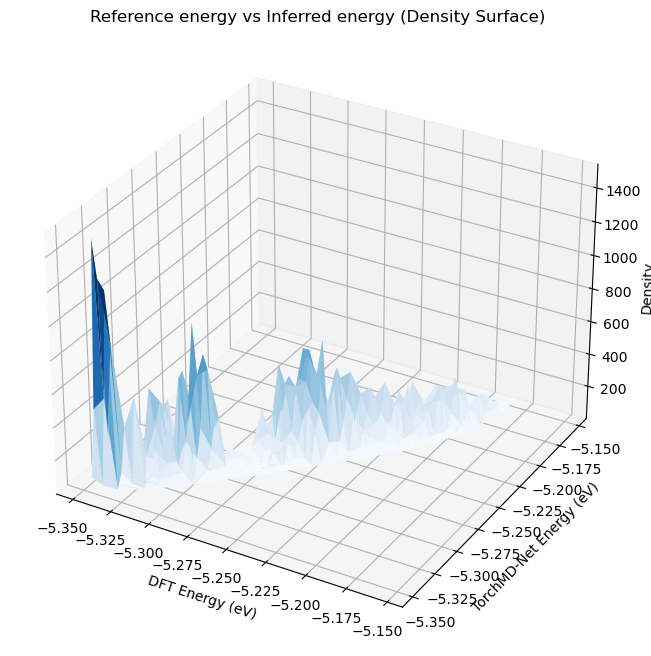
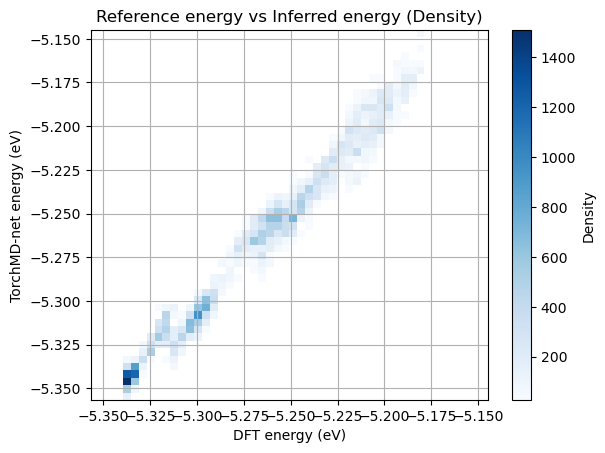
seed: 42

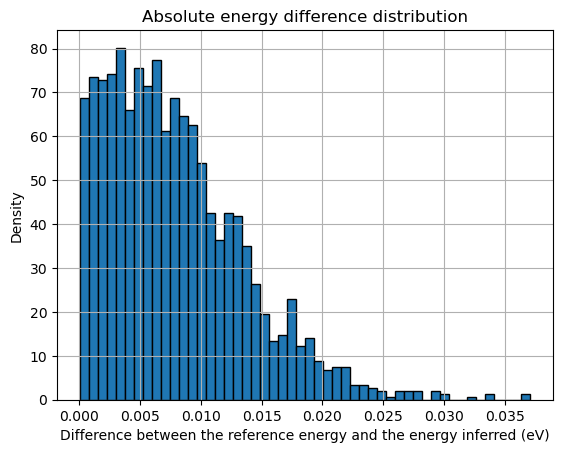
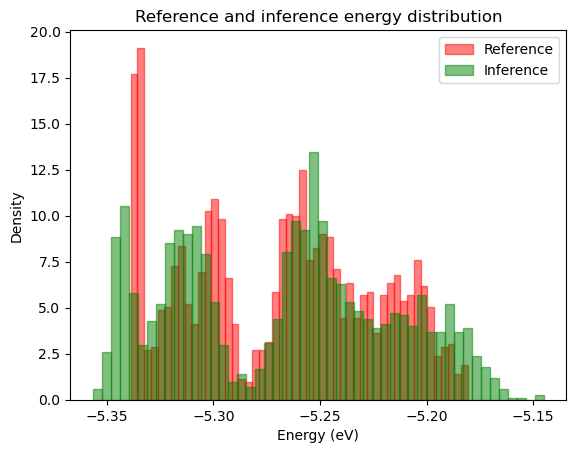
test\_size: 0.1

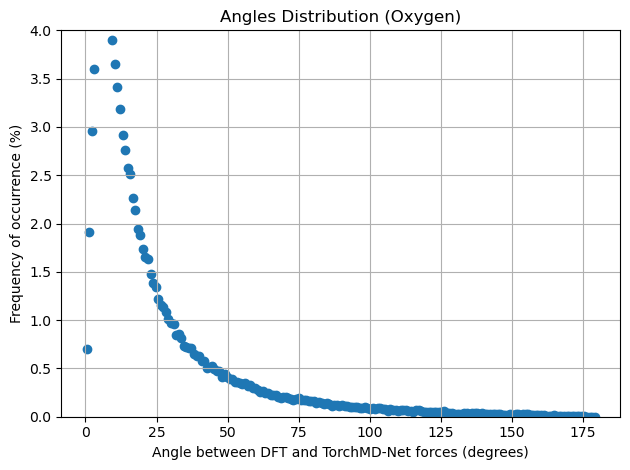
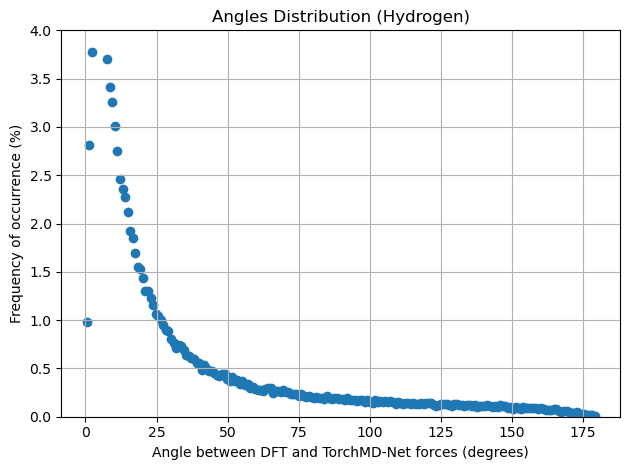
train\_size: 0.81

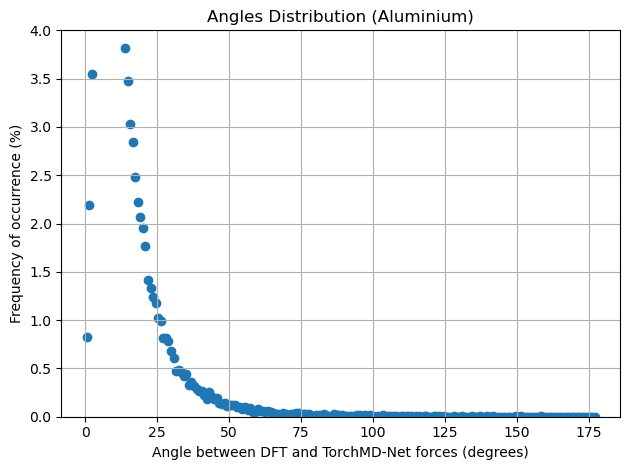
val\_size: 0.09

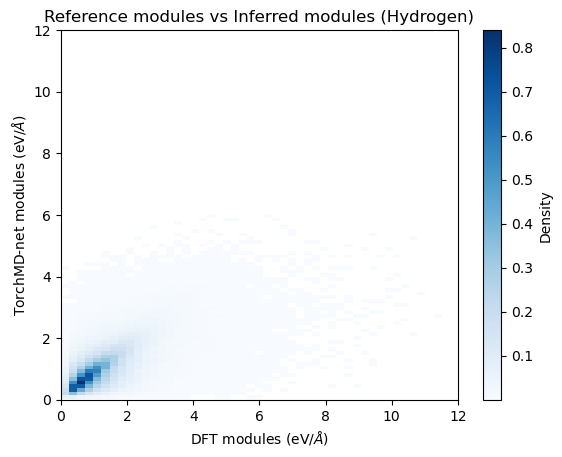
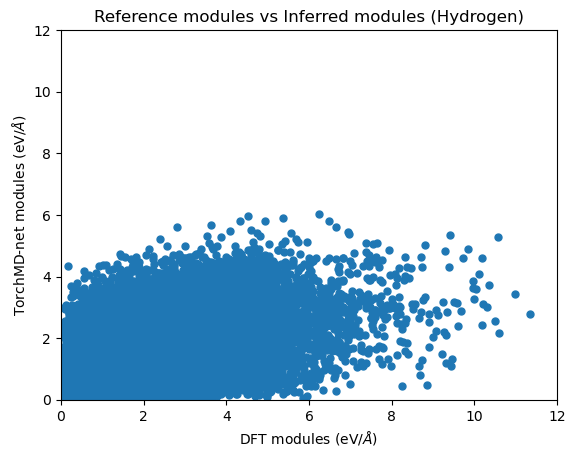


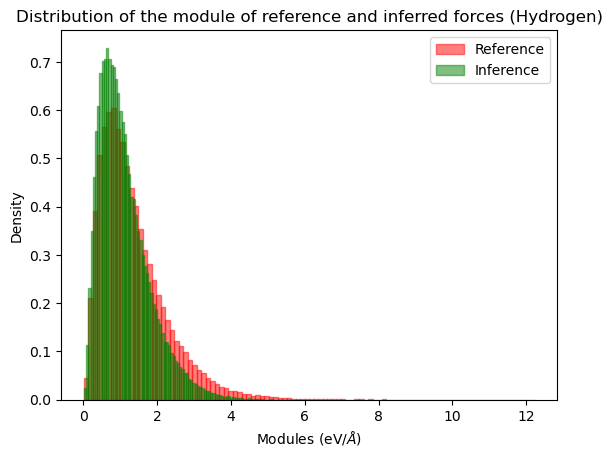
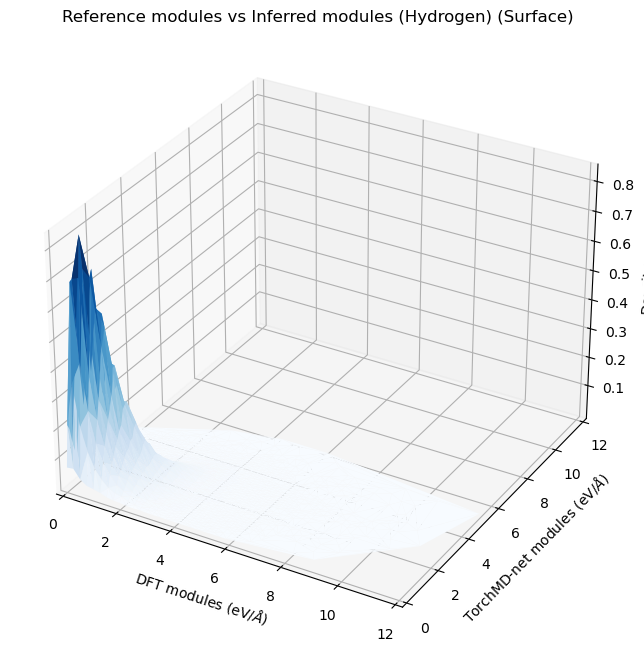


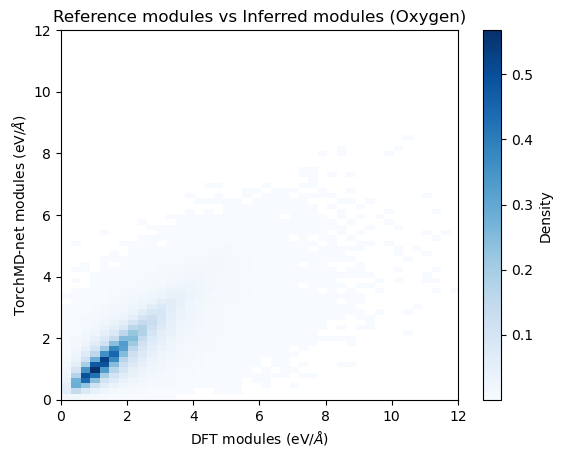
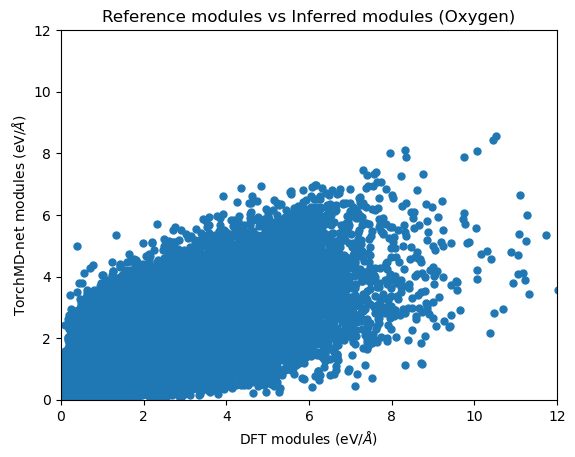


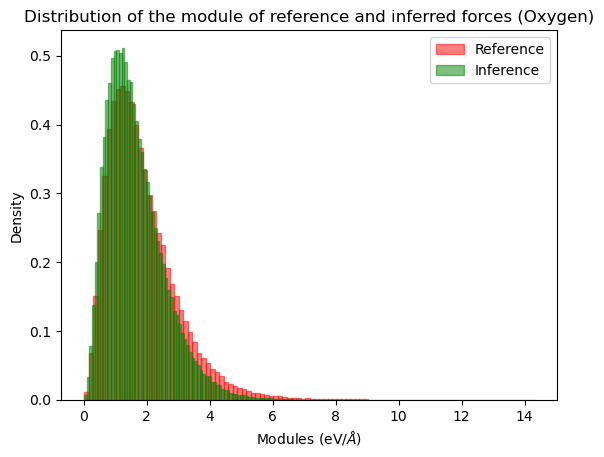
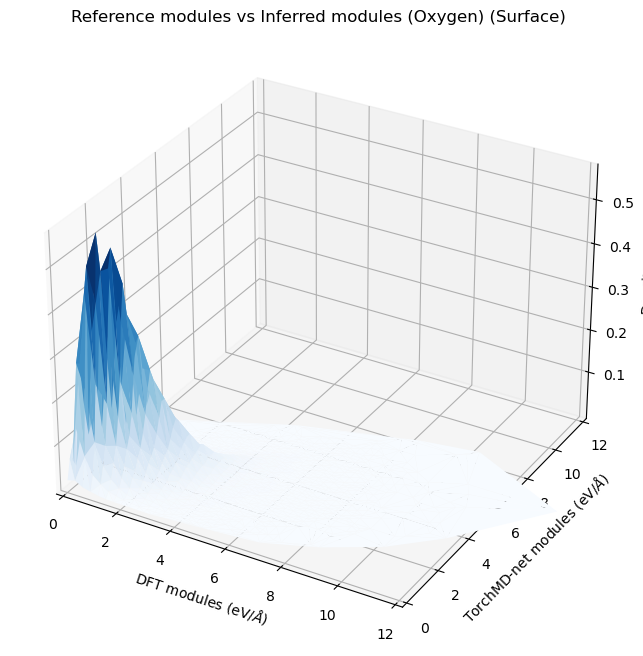


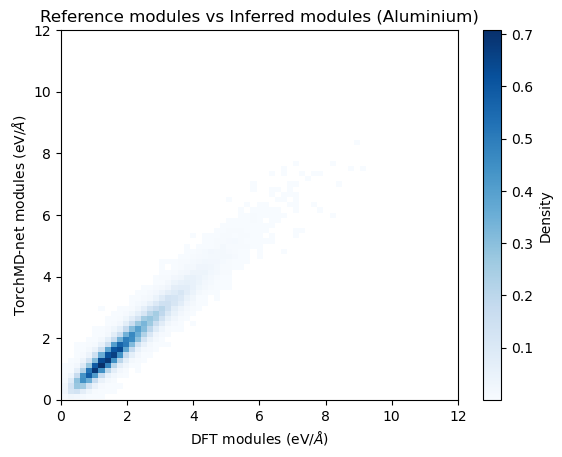
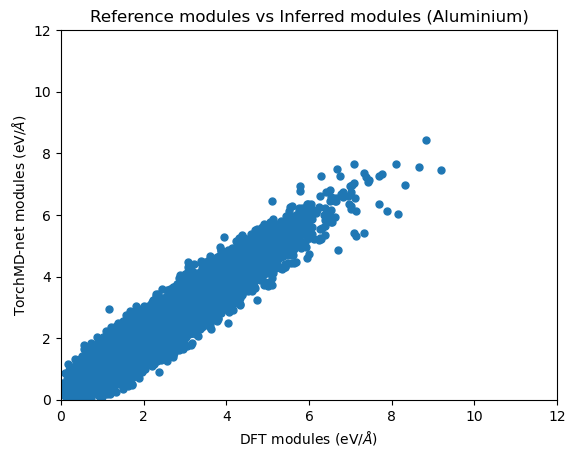


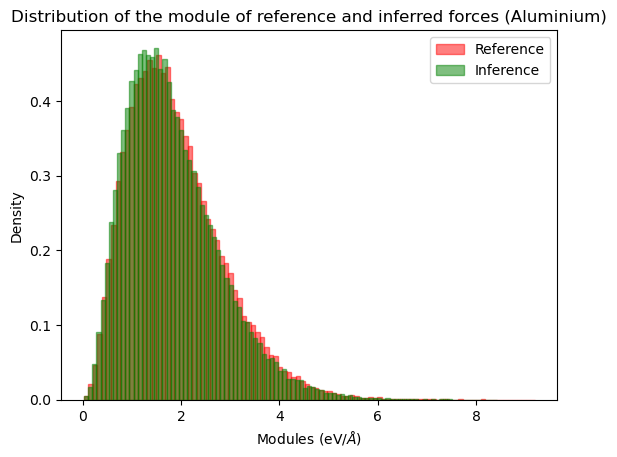
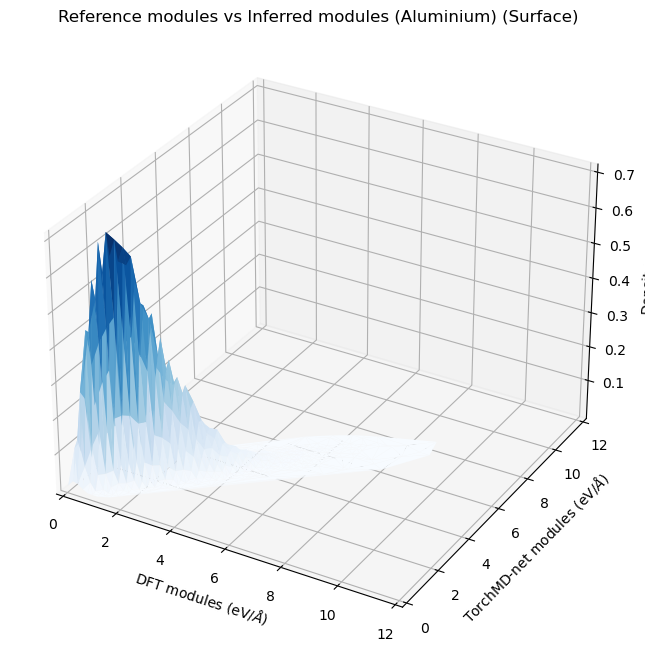












# Conclusiones

La arquitectura es poco sensible a la adición de fuerzas. Cuando se le da el mismo peso a los dos targets, la energía y las fuerzas, el resultado del entrenamiento es prácticamente igual al que no tiene fuerzas. Por lo que se tiene que disminuir mucho el peso del target energía para que el target fuerzas pueda incidir sobre el modelo.

Cuando las fuerzas tienen considerablemente más peso que la energía se puede apreciar su efecto, sin embargo esta mejora en la inferencia de las fuerzas es a costa de la precisión de las inferencias de la energía y aun así los resultados de las fuerzas no son satisfactorios.

El modelo TorchMD-Net es bueno calculando la energía del arreglo a traves del tiempo sin embargo sus resultados con fuerzas deja mucho que desear en todos los casos.

Aun hay un area de oportunidad, la modificación de los hiperparametros puede ser un factor decisivo ya que se puede adecuar buena parte de la arquitectura a resolver un problema en particular.