

INFORME PROGRÉS 1

1 INTRODUCCIÓ:

En aquest segon informe de progrés, es detalla l'evolució del projecte en qüestió, el qual ha experimentat canvis imprevistos a causa de la situació documentada anteriorment sobre la publicació de SchNetPackV2. Aquests imprevistos han obligat a adoptar una nova direcció en el projecte, amb l'establiment de nous objectius. Malgrat aquests canvis, s'ha mantingut la mateixa metodologia descrita en l'informe inicial, amb adaptacions realitzades a la fase 4, i la decisió de descartar la fase 5, que inicialment es considerava opcional.

2 PLANIFICACIÓ V2:

La nova planificació de la fase 4 del projecte consisteix en l'optimització dels models proporcionats per la toolbox SchNetPackv2 sobre la base de dades QM9, QM9 és una base de dades química que s'ha convertit en un referent en el camp de la Dinàmica Molecular (DM) i l'aprenentatge automàtic aplicat a la DM. Prové de l'estudi teòric i computacional de molècules orgàniques de baixa massa, i conté informació detallada sobre diverses propietats químiques clau.

En la primera fase, ens centrarem en l'ajust dels hiperparàmetres estrictament computacionals. Aquests hiperparàmetres, com la taxa d'aprenentatge (learning rate), el nombre d'èpoques d'entrenament (epochs) i la mida del lot (batch size), tenen un impacte directe en el procés d'entrenament i afecten el rendiment del model. Mitjançant la cerca i l'optimització d'aquests atributs explícitament computacionals, buscarem aconseguir el millor rendiment possible dels models.

En la segona fase, canviarem el focus de l'estudi als paràmetres interns de SchNetPack, com la llista de veïns neighbor list i els paràmetres dels models de representació entre d'altres.

Per dur a terme aquestes fases d'optimització, s'utilitza la plataforma Wandb, que proporciona eines eficaces per gestionar i monitorar els experiments d'aprenentatge automàtic. Amb Wandb, podem visualitzar els resultats, comparar diferents experiments, fer un seguiment de mètriques clau i emmagatzemar models i resultats per a un accés posterior. Això ens permetrà realitzar proves iteratives i ajustar els hiperparàmetres i paràmetres interns dels models SchNetPackv2 amb l'objectiu d'aconseguir el millor rendiment possible.

Així doncs els nous objectius de la fase 4 queden definits en la següent taula.

Objectiu	Descripció	Prioritat
O4.1	Reforçar coneixements del objectiu O3.1 (estudi de les bases de dades usades en el camp)	Essencial
O4.2	Realitzar estudi de hiperparàmetres computacionals.	Essencial
	O4.2.1 Paràmetres batch_size i epochs	
	O4.2.2 Paràmetre learning rate (lr)	
O4.3	Realitzar estudi dels mòduls de propietats físiques	Essencial

3 ASSOLIMENTS:

En aquest apartat tractarem el progrés realitzat en els objectius de la última fase del projecte, incloent els objectius O4.1, O4.2 i els inicis de O4.3. També es tractarà en part la metodologia usada però no en el mateix format que l'anterior informe de progrés.

3.1 BASE DE DADES:

En aquesta fase del projecte s'ha decidit fixar una base de dades sobre la qual fer tots els experiments, aquesta es la base de dades QM9^[4]. La base de dades QM9 és una col·lecció de dades moleculars que proporciona informació essencial sobre propietats químiques i físiques de diverses molècules orgàniques petites. Aquesta base de dades s'ha convertit en una font de referència en els camps de la química computacional i la descoberta de materials.

Les dades de la base de dades QM9 provenen de simulacions quàntiques utilitzant mètodes tradicionals (completar amb ajuda del Jordi). Les simulacions s'han dut a terme per a diverses molècules orgàniques amb un nombre limitat d'àtoms. Aquestes simulacions proporcionen informació detallada sobre diverses propietats moleculars, com ara energies, capacitats calorífiques, dipols moleculars, polaritzabilitats, entre altres.

La base de dades QM9 empra el format de dades ASE (Atomic Simulation Environment) per emmagatzemar les seves entrades. ASE és una llibreria de Python amplament utilitzada per a càlculs i anàlisis de simulacions atòmiques i moleculars. El format ASE proporciona una estructura consistent i flexible per emmagatzemar informació sobre àtoms, molècules i càlculs associats.

Cada entrada de la base de dades es un objecte de tipus `AtomsRow` en la base de dades te un numero de keys personalitzable però sempre hi ha les següents 9:

Taula 1: Claus presents en cada entrada de la base de dades QM9. Taula extreta de la web oficial de ASE^[8].

key	descripció	tipus de dada	forma
id	Local database id	int	
unique_id	Globally unique hexadecimal id	str	
ctime	Creation time	float	
mtime	Modification time	float	
user	User name	str	
numbers	Atomic numbers	int	(N,)
pbcb	Periodic boundary condition flags	bool	(3,)
cell	Unit cell	float	(3, 3)
positions	Atomic positions	float	(N, 3)

A part de això

3.2 ESTUDI DE HIPERPARÀMETRES COMPUTACIONALS:

En la primera fase de l'experiment, ens centrem en l'ajust dels hiperparàmetres estrictament computacionals amb l'objectiu de millorar el rendiment dels nostres models. Aquesta fase és fonamental ja que els hiperparàmetres, com la taxa d'aprenentatge, el nombre d'èpoques d'entrenament i la mida del lot, tenen un impacte directe en el procés d'entrenament i poden afectar significativament el rendiment final del model.

Utilitzem la plataforma de monitoratge i optimització de models, wandb (Weights & Biases). Wandb ens permet explorar i ajustar els hiperparàmetres de manera sistemàtica fent l'ús de Sweeps, mantenir un seguiment de les execucions d'entrenament i analitzar els resultats per prendre decisions informades.

3.2.1 Paràmetres `batch_size` i `epochs`:

Ens hem centrat principalment en dues variables clau: `batch_size` i `epochs`. El nombre d'èpoques determina quantes vegades el model passa per tot el conjunt de dades d'entrenament, mentre que la mida del lot especifica la quantitat de mostres que es processen en cada pas d'actualització dels pesos del model.

Per dur a terme la cerca i l'optimització dels hiperparàmetres, hem utilitzat la funcionalitat de Sweeps de wandb. Amb aquesta característica, hem pogut definir un espai de cerca que inclou diferents valors per als hiperparàmetres de les èpoques i la mida del lot. Wandb s'encarrega de generar combinacions úniques de valors dins d'aquest espai i executa múltiples execucions d'entrenament per a cada combinació.

3.2.1.1 Paràmetres

El `sweep_configuration` s'utilitza per indicar al sweep com ha de actuar, es defineix; els possibles valors que cada paràmetre pot prendre, quina mètrica es l'objectiu, si es vol minimitzar o maximitzar aquesta, etc. En aquest experiment `batch_size` i `epochs` estan fixats en una llista de possibilitats metres que `lr` es un rang per donar la màxima flexibilitat perquè els paràmetres que volem estudiar siguin els principals afectats. Fixar el valor de `lr` produiria una desavantatge per a convencions de paràmetres, per exemple si estem provant un `batch size` de 4096 i fem 10 epochs amb un valor de `lr` molt baix, el model no tindrà temps per aprendre.

```
sweep_configuration = {
    'method': 'random',
    'name': 'sweep',
    'metric': {
        'goal': 'minimize',
        'name': 'val_loss'
    },
    'parameters': {
        'batch_size': {'values': [512, 1024, 1536, 2048, 2560, 3072, 3584, 4096]},
        'epochs': {'values': [8, 10, 12, 14, 16, 18, 20]},
        'lr': {'max': 0.01, 'min': 0.0005}
    }
}
```

Figura 1: Fragment del codi utilitzat per configurar el sweep. El sweep haurà de escollir de manera aleatòria entre els valors establerts en l'apartat de paràmetres, per a `epochs` i `batch_size` podrà escollir els valors dintre les llistes definides mentre que per el valor de `lr`, podrà triar el valor en el rang 0.01 fins a 0.0005.

3.2.1.2 Resultats:

Tots els resultats exposats provenen d'un conjunt de 6 experiments diferents, els quals sumen un total de 65 models entrenats.

Com mencionat anteriorment s'utilitza la plataforma Wandb per representar i monitoritzar els models. Mitjançant l'eina de report s'ha construït una sèrie de gràfics per a simplificar i sintetitzar els resultats. Les representacions per defecte (figura 2) no aporten massa informació, es necessari definir un mateix els gràfics que es busquen per a poder aconseguir informació més polida.

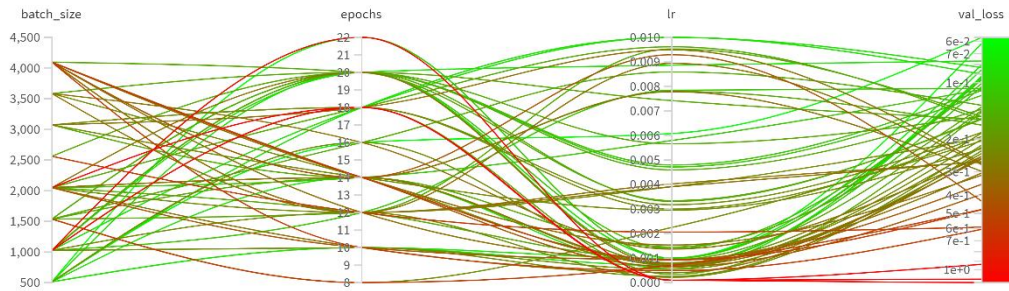


Figura 2: Gràfic predeterminat proporcionat per Wandb per representar un sweep. Agrupa tots els models, cada un és representat per una línia que passa pels paràmetres que ha escollit i acaba al valor de val_loss que obté, aquest li dona un color obtingut d'un gradient definit per l'usuari.

Un dels apartats més importants del report de l'experiment es la correlació entre els paràmetres i la val_loss (mètrica objectiu). En aquest apartat, podem visualitzar en verd/vermell si la correlació es positiva o negativa respectivament, això implica una correlació directe o inversa. Tenint en compte que busquem minimitzar el valor de val_loss, partint de la Figura 3 podem inferir que molt probablement per batch_size serà millor mantenir els valors baixos i que per epochs serà el contrari, valors alts donaran millors resultats.

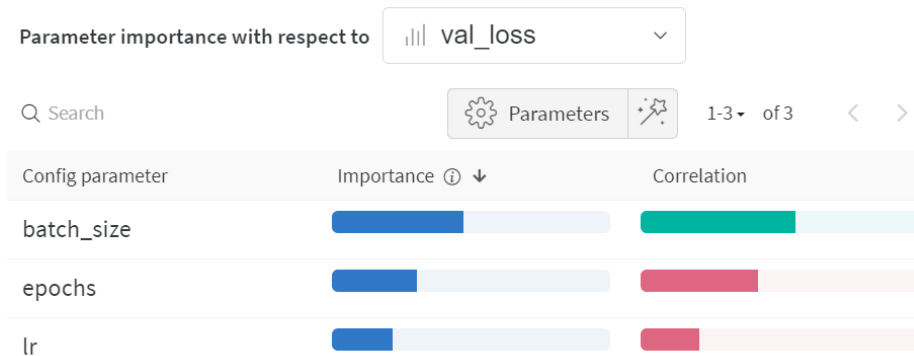


Figura 3: Representació de la correlació entre els paràmetres i la mètrica de sortida val_loss, es pot apreciar que batch_size te una correlació positiva (en verd) i epochs la te negativa (en vermell).

Es una evidència bastant solida però per assegurar aquestes assumpcions es representaran les dades amb un altre mètode mes adequat. S'utilitzaran tres gràfiques dedicades per a cada paràmetre per a poder aïllar-lo de la resta.

Teoria 1: Valors de batch_size petits i numero de epochs grans minimitzarà el val_loss.

La idea principal per a les segones gràfiques es agrupar els models per la mètrica que es vol estudiar, per després seleccionar els grups mes rellevants i poder visualitzar clarament la importància de cada mètrica per al resultat final.

3.2.1.2.1 Batch_size:

S'han escollit els valors 512, 2048 i 4096 (valors petit, mitja i gran). Aquests valors representen el rang de valors possibles a escollir molt adequadament i permeten visualitzar les teories comentades en la figura 3.

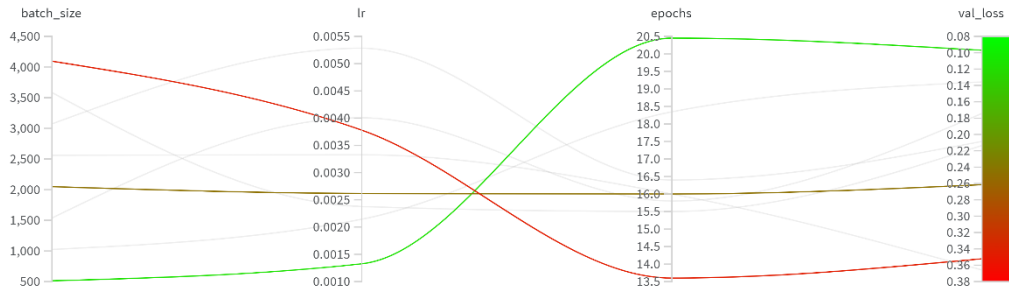


Figura 4: Gràfic base dels paràmetres d'un sweep. S'observa clarament la correlació negativa que té batch_size amb val_loss. Valors més grans de batch_size porten a val_loss grans ≈ 0.35 valors més petits (512) a valors petits ≈ 0.10 , queda representat per el color de les línies (vermell gran, verd petit).

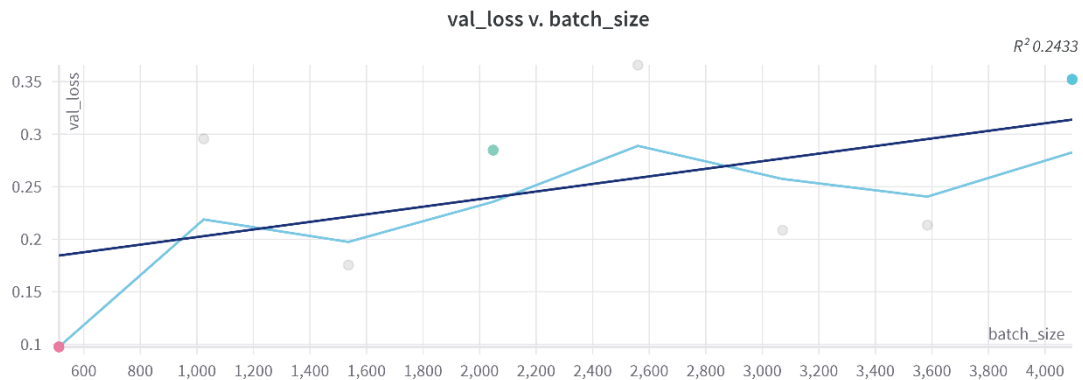


Figura 5: Gràfic custom amb batch_size en l'eix de les x's i val_loss en el de les y's. Cada punt representa la mitja de val_loss del batch_size en el que es troba. La línia en blau fosc és la regressió lineal amb pendent 0.2433 (indicat en la cantonada superior dreta). De nou es pot interpretar clarament que per valors més grans de batch_size obtenim valors grans de val_loss, s'aplica també a valors petits.

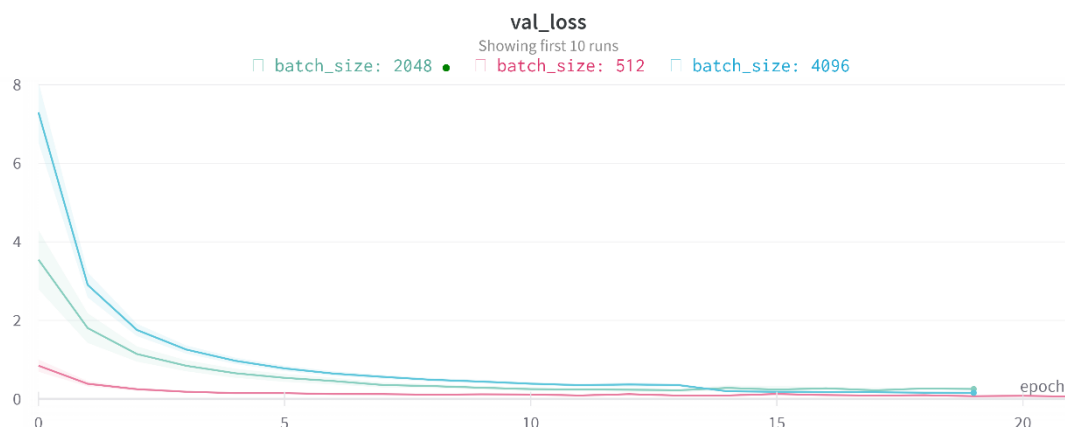
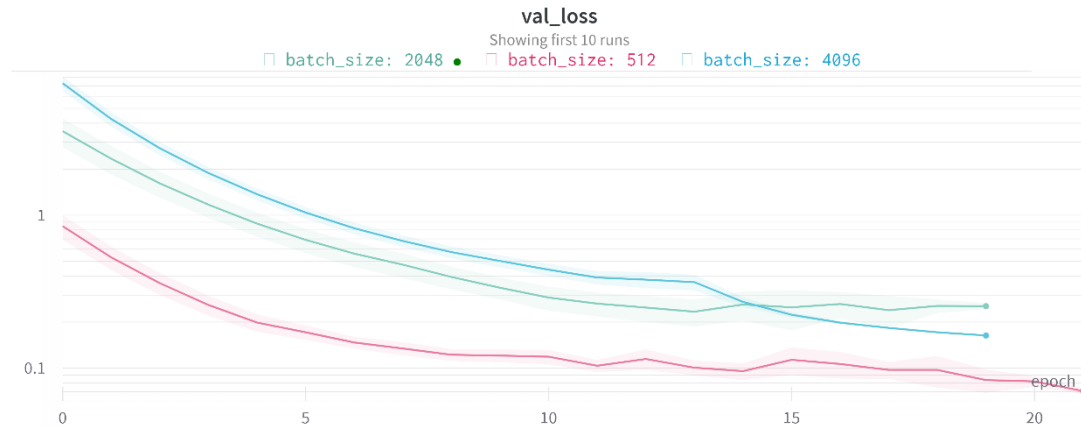


Figura 6: Gràfica de la evolució temporal de val_loss durant les epochs de l'entrenament dels models. Podem veure com per valors de batch_size petits la corba s'estabilitza en un valor de traint_loss amb un numero de epochs molt menor que la resta de batch_size. Com més gran es la batch_size més tarda en arribar a estabilitzar-se. Per visualitzar millor els valors petits als finals de les corbes es defineix la Figura 7.



Teoria 2: Els valors de batch_size grans donen mals resultats perquè els falta temps/epochs per entrenar. Si augmenta el numero de epochs la val_loss del model disminuirà.

3.2.1.2.2 Epochs:

S'escull fer les representacions dels següents gràfics fent l'ús de 4 valors, donat que els models tenen més opcions l'hora de escollir el numero de epochs.

Els valors escollits són 8, 12, 18 i 22.

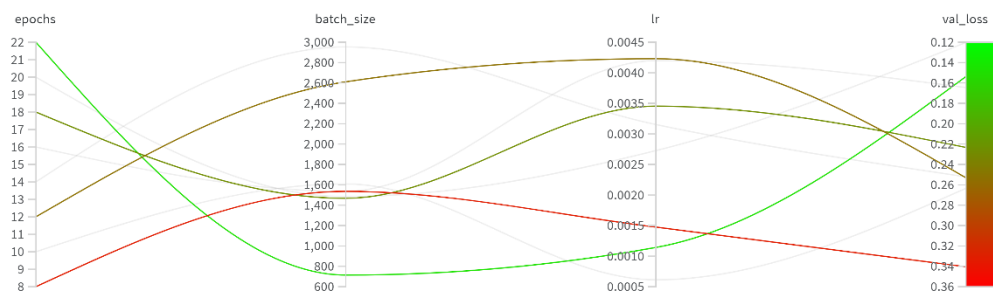
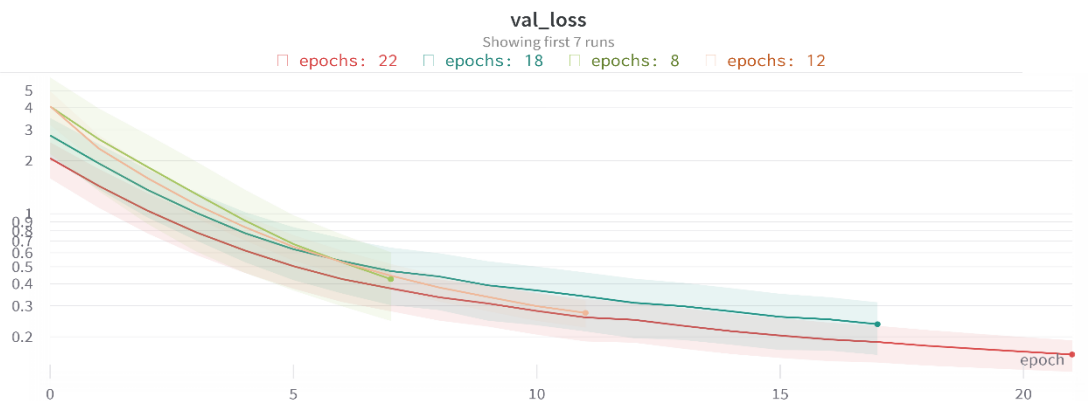
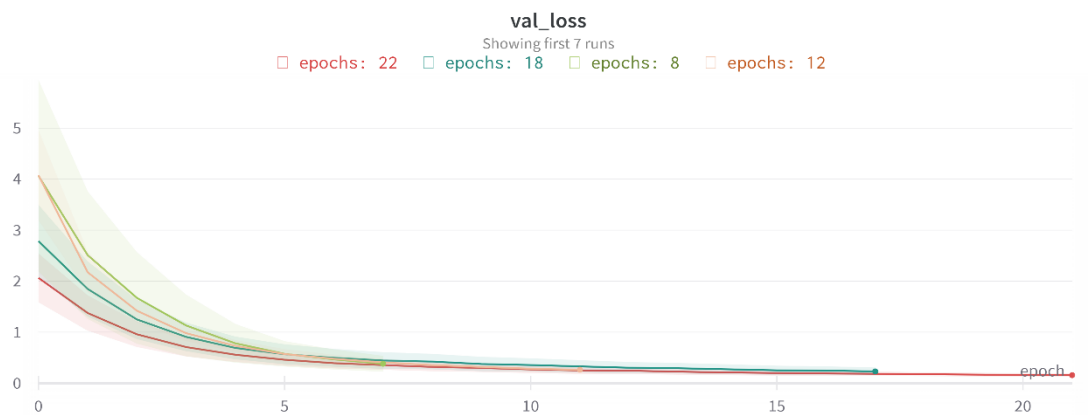
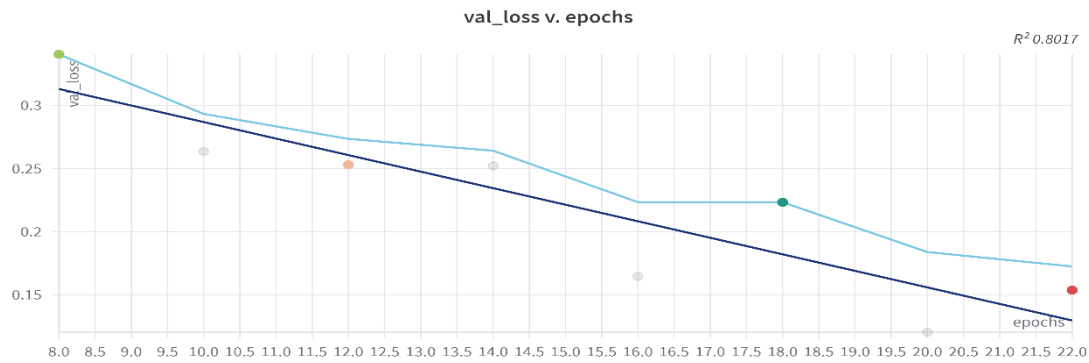


Figura 8: Gràfic base dels paràmetres d'un sweep. S'observa clarament la correlació negativa que té epochs amb val los. Valors més petits de epochs porten a val_loss grans ≈ 0.35 i valors més grans a valors petits ≈ 0.15, queda representat per el color de les línies (vermell gran, verd petit).



Teoria 1: Valors de `batch_size` petits i numero de `epochs` grans minimitzarà el `val_loss`.

Donades les evidències en les Figures 3-11 podem determinar aquesta teoria com a certa. Queden establerts els valors:

Batch_size: 512

Epochs: 22

3.2.1.2.3 Learning rate (lr):

Per a l'estudi del valor de lr normalment es fa l'ús de potències de 10, tot i això en els experiments realitzats per als paràmetres de `batch_size` i `epochs` es va usar un rang des de 0.01 fins a 0.0005, per a seguir els estàndards del camp s'ha realitzat un nou experiment amb 20 models en el que només s'usen els valors 0.01, 0.001 i 0.0001. En aquest s'ha fixat els valors de `epochs` entre 18 i 36 i `batch_size` entre 512 i 2048 per a permetre que tots els valors de lr disposin de paràmetres més adequats. Tal i com s'ha fet en els altres paràmetres estudiats, s'han agrupat tots els models per lr utilitzat per a simplificar les gràfiques.

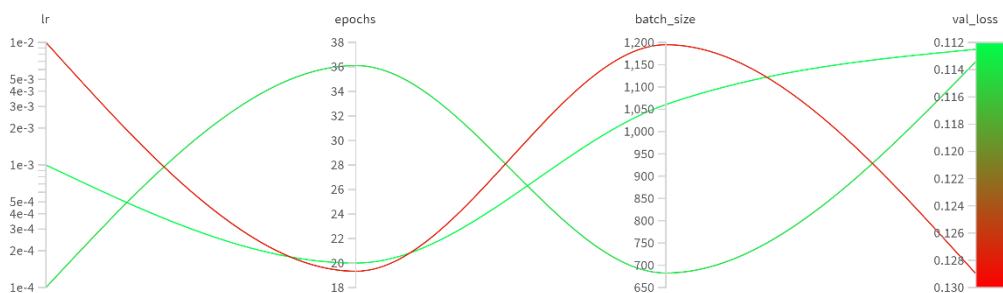


Figura 12: Gràfic base dels paràmetres d'un sweep. L'eix de lr està en escala logarítmica donat que el rang de valors que pren no és lineal. S'observa que mentre que els valors de lr 0.001 i 0.0001 queden prou igualats, el valor de 0.01 es desmarca (negativament) per un marge considerable.

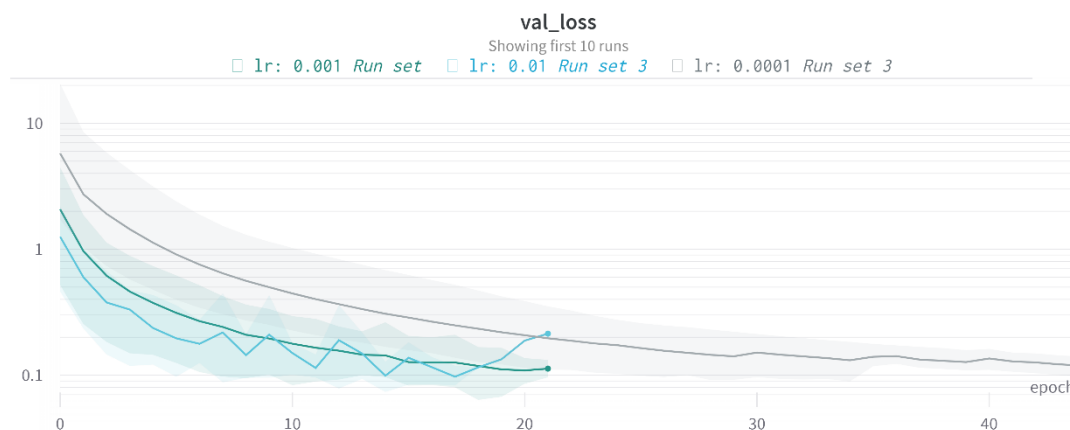


Figura 13: Gràfica de la evolució temporal de `val_loss` durant les `epochs` de l'entrenament dels models. L'eix de les y's està en una escala logarítmica per visualitzar millor la etapa d'estabilització dels models en el que `val_loss` varia molt més lent. S'observen 3 conceptes importants, els models entrenats amb lr de 0.01 arriben molt ràpid a "estabilitzar-se" on comencen a incrementar i decrementar el valor de `val_loss` molt ràpid denotant que és un valor de lr massa gran, els models amb valor de lr 0.0001 tarden massa en arribar al seu mínim de `val_loss`, els models que usen lr 0.001 són el mixt perfecte entre els dos anteriors, arriben ràpid al mínim i s'estabilitzen sense variacions de `val_loss` extremes.

Donades les evidències en les Figures 12 i 13 podem establir el valor òptim de learning rate:

lr: 0.001

3.2.1.2.4 Test amb èpoques extenses:

Per a aprovar/denegar la teoria 2 es fa un experiment on s'usa la sweep config de la figura 14.

```
sweep_configuration_lr = {
  'method': 'grid',
  'name': 'Sampling',
  'metric': {
    'goal': 'minimize',
    'name': 'val_loss'
  },
  'parameters': {
    'batch_size': {'values': [512, 1024]},
    'epochs': {'values': [30, 35, 40, 45]},
    'lr': {'values': [0.0001]},
    'trainingCutoff': {'values': [5]},
    'dataCutoff': {'values': [5]},
    'n_atom_basis': {'values': [30]}
  }
}
```

Figura 14: Fragment del codi utilitzat per configurar el sweep. El sweep iterarà per totes les configuracions possibles entre els paràmetres indicats.

4 CONCLUSIONS:

5 OBSERVACIONS:

6 BIBLIOGRAFIA:

- [1] - K. T. Schütt, P. Kessel, M. Gastegger, K. A. Nicoli, A. Tkatchenko, and K.-R. Müller. **SchNetPack: A Deep Learning Toolbox For Atomistic Systems**. *Journal of chemical theory and computation*, 2019, 15, 448-455.
- [2] - Stefan Doerr, Maciej Majewski, Adrià Pérez, Andreas Krämer, Cecilia Clementi, Frank Noe, Toni Giorgino, and Gianni De Fabritiis. **TorchMD: A Deep Learning Framework for Molecular Simulations**. *Journal of chemical theory and computation*, 2021, 17, 2355–2363.
- [3] - MD17 (Molecular Dynamics 17): <https://paperswithcode.com/dataset/md17> (accessed 5/3/2023).
- [4] - Molecular Property Prediction on QM9: <https://paperswithcode.com/sota/molecular-property-prediction-on-qm9> (accessed 05/3/2023).
- [5] - Takeru Miyagawa, Kazuki Mori, Nobuhiko Kato, Akio Yonezu. **Development of neural network potential for MD simulation and its application to TiN**. *Computational Material Science*, 15 April 2022, 111303.
- [6] - Falcon, W. PyTorch Lightning. 2019, GitHub. <https://github.com/PyTorchLightning/pytorch-lightning> (accessed 24/3/2023).
- [7] - Kristof T. Schütt, Stefaan S. P. Hessmann, Niklas W. A. Gebauer, Jonas Lederer, Michael Gastegger; **SchNetPack 2.0: A neural network toolbox for atomistic machine learning**. *J. Chem. Phys.* 14 April 2023; 158 (14): 144801