1. Jika menggunakan model MLP dengan 3 hidden layer (256-128-64) menghasilkan underfitting pada dataset ini, modifikasi apa yang akan dilakukan pada arsitektur? Jelaskan alasan setiap perubahan dengan mempertimbangkan bias-variance tradeoff!
2. Selain MSE, loss function apa yang mungkin cocok untuk dataset ini? Bandingkan kelebihan dan kekurangannya, serta situasi spesifik di mana alternatif tersebut lebih unggul daripada MSE!
3. Jika salah satu fitur memiliki range nilai 0-1, sedangkan fitur lain 100-1000, bagaimana ini memengaruhi pelatihan MLP? Jelaskan mekanisme matematis (e.g., gradien, weight update) yang terdampak!
4. Tanpa mengetahui nama fitur, bagaimana Anda mengukur kontribusi relatif setiap fitur terhadap prediksi model? Jelaskan metode teknikal (e.g., permutation importance, weight analysis) dan keterbatasannya!
5. Bagaimana Anda mendesain eksperimen untuk memilih learning rate dan batch size secara optimal? Sertakan analisis tradeoff antara komputasi dan stabilitas pelatihan!

Jawab dan analisis:

1. Beberapa modifikasi yang bisa dilakukan:
   - Menambahkan hidden layer dari (256-128-64) menjadi (512-256-128-64):
     Karena menambah neuron dan layer meningkatkan kompleksitas model sehingga mengurangi bias, cocok untuk dataset besar dan kompleks (515K sampel, 90 fitur pada dataset RegresiUTSTelkom).
   - Mengganti Relu biasa ke LeakyRelu untuk menghindari dying Relu
   - Menambahkan dropout (0.2-0.5) untuk menghindari overfitting dan meyeimbangkan variance.
   - Menambahkan EarlyStopping & regulasi L2 untuk mengendalikan variance.

2. Alternatif lain selain MSE dengan kelebihan dan kekurangannya:

Loss Function	Kelebihan	Kekurangan	Cocok untuk?
MAE (Mean Absolute Error)	Robust terhadap outlier	Gradien konstan mengakibatkan lambat konvergen	Data dengan noise tinggi
Huber Loss	Kombinasi MSE & MAE (kuadrat untuk error kecil, absolut untuk besar)	Butuh parameter δ (threshold)	Stabil terhadap outlier dan tetap halus untuk optimisasi
Log-Cosh Loss	Hampir seperti MSE, tapi lebih tahan outlier	Lebih lambat dihitung	Alternatif lembut dari MSE

Singkatnya, jika banyak outlier, Huber atau MAE lebih unggul daripada MSE yang sangat sensitif terhadap error besar.

3. Efek negatifnya:
   - Gradien tidak seimbang saat backpropagation:
     - Fitur dengan skala besar (100–1000) mendominasi perhitungan dot product: $z = wTx + b$
     - error akan lebih besar untuk fitur berskala besar sehingga update weight tidak seimbang
   - Vanishing Gradient (Terjadi saat layer-layer dalam mendapat z besar setelah aktivasi (ReLU, LeakyReLU), banyak output bisa mendekati nol sehingga turunannya juga mendekati nol dan update kecil terus-menerus (model stuck))
   - Convergence lambat (Gradien error jadi tidak sinkron sehingga optimisasi jadi melambat karena tidak semua bobot learning dengan cepat)

   Secara matematis:

   Misal fitur: $x1$ = 0.5, $x2$ = 500

   Bobot awal: w = [0.01,0.01]

   Dot product: z = 0.01∗0.5 + 0.01∗500 = 5.005

   Sehingga fitur kedua mendominasi hasil prediksi dan gradien.

   Solusinya yaitu dengan malakukan normalisasi fitur.

4. Penjelasan metode teknikal untuk mengukur kontribusi relatif setiap fitur tanpa mengetahui namanya:
   1. Permutation Feature Importance: acak nilai satu fitur, ukur degradasi performa (misal kenaikan MAE).
      - Kelebihannya agnostik terhadap model.
      - Kekurangannya lambat, tidak cocok jika fitur saling korelasi tinggi.
   2. Weight Magnitude Analysis (hanya untuk linear layer): amati nilai absolut bobot dari input ke hidden layer.
      - Kelemahannya tidak akurat untuk model deep/non-linear.
   3. SHAP (SHapley Additive exPlanations): metode game theory, kontribusi setiap fitur terhadap setiap prediksi.
      - Kuat, tapi mahal secara komputasi.
   4. Partial Dependence Plot (PDP): Visualisasi bagaimana nilai fitur tertentu memengaruhi prediksi.

5. Pertama bisa dengan cara search menggunakan gridsearch atau randomsearch ataub optuna, misal:
   LR: [1e-4, 1e-3, 1e-2]
   Batch Size: [32, 64, 128, 256]

   Lalu analisis loss atau accuracy atau error dan kecepatan train per epochnya.

   Menurut saya LR 1e-4 serta batch size 32/64/128 sudah sangat bagus untuk standar spek jalan di lokal yang limitasinya RAM dan VRAM. Jika model kompleks dan dataset besar (mamakan banyak

RAM/VRAM) bisa menggunakan batch size 32 karena batch size yang lebih besar juga menggunakan RAM/VRAM besar juga.