Machine Learning

|Early Stopping

Prof. Dr. Mostafa Elhosseini

https://youtube.com/@ProfElhosseiniTechVerse

Agenda

- Early stopping

Early stopping

- Regularization directly **modifies the learning algorithm** to reduce the **complexity** of the model.

- It does this by **adding a penalty term** to the loss function, which the training process aims to minimize

Early stopping

- Early stopping **doesn't alter** the model itself or its loss function. Instead, it **monitors the model's performance** on a **validation dataset** during the training process.

- **Training is halted** when the model's performance on the validation dataset starts deteriorating, despite potential improvements on the training dataset

- It is such a simple and efficient regularization technique

Early stopping

- Early stopping **doesn't alter** the model itself or its loss function. Instead, it **monitors the model's performance** on a **validation dataset** during the training process.

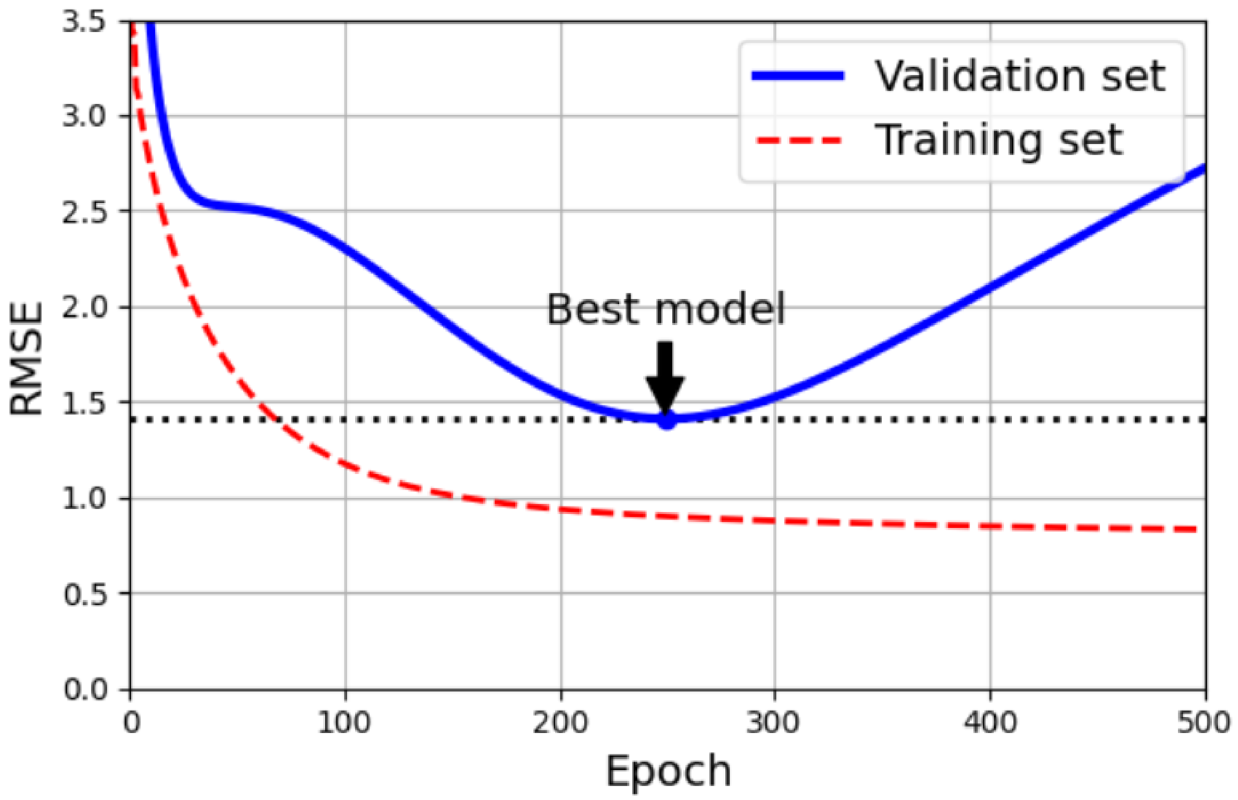
- **Training is halted** when the model's performance on the validation dataset starts deteriorating, despite potential improvements on the training dataset

- Early stopping ensures that the model **does not waste time** learning from noise or irrelevant details in the training data, potentially leading to better performance on new, unseen data.

How It Works:

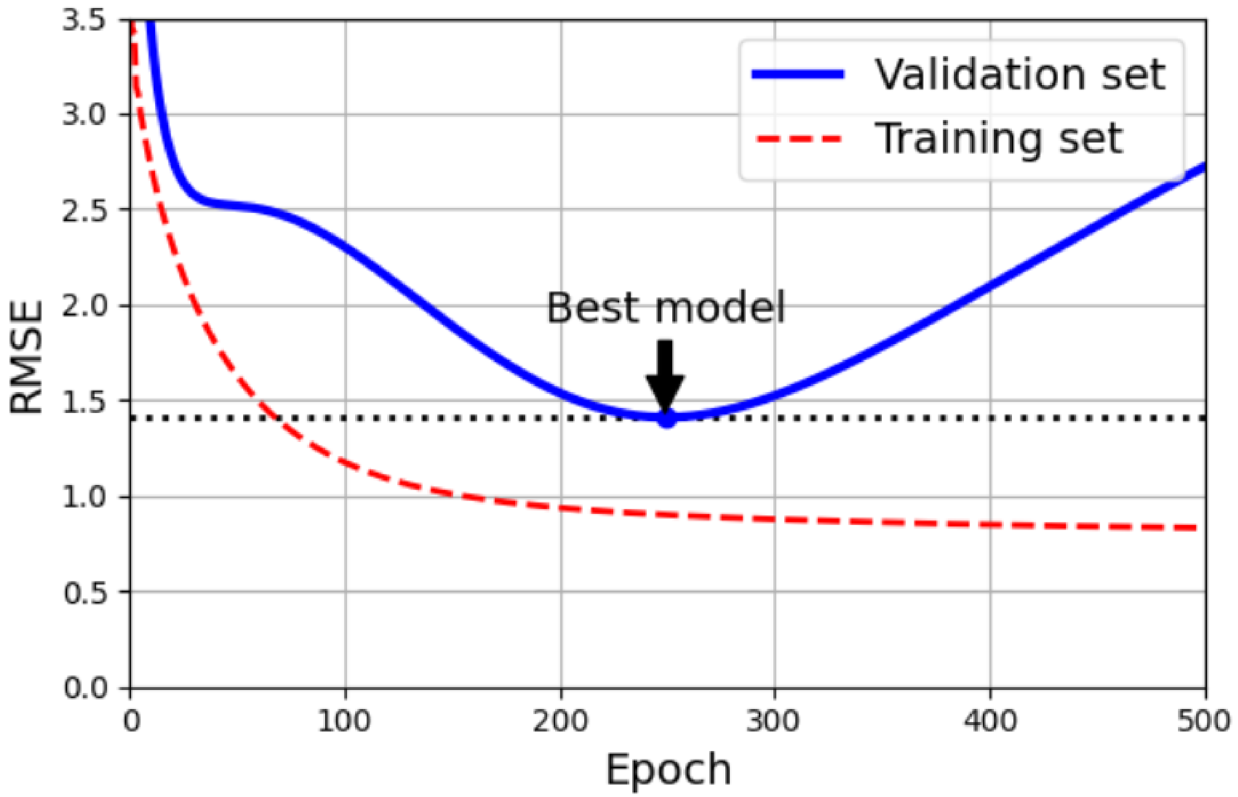
- Train the model for a number of epochs.

- Monitor the model's performance on both the training set and a separate validation set.



How It Works:

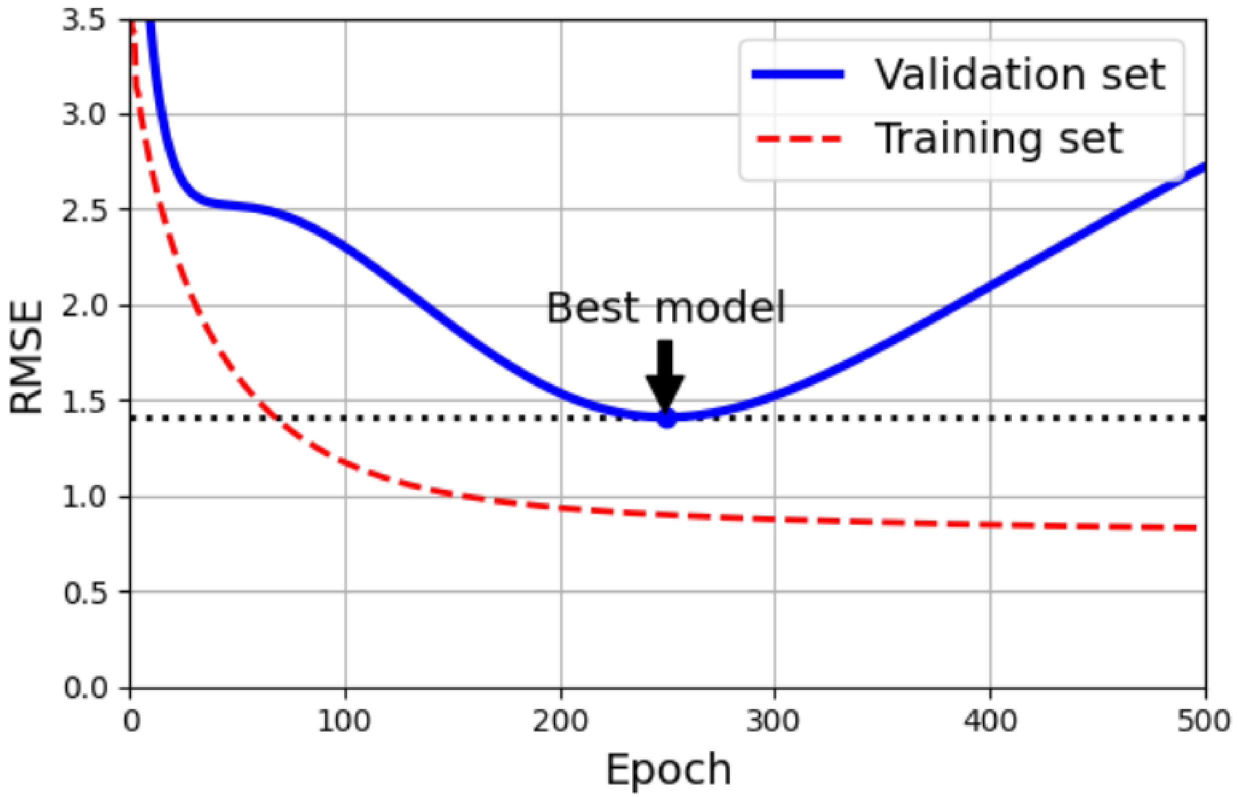
- Identify when the validation error stops decreasing and begins to increase, indicating the onset of overfitting.
- Stop training at this point, ideally reverting to the model state where the validation error was at its lowest.



How It Works:

- Identify when the validation error stops decreasing and begins to increase, indicating the onset of overfitting.

- Stop training at this point, ideally reverting to the model state where the validation error was at its lowest.



Advantages

- Simple to implement.
- Often leads to better generalization on unseen data.
- Saves computational resources by reducing unnecessary training time.

Stochastic and mini-batch gradient descent

- **Stochastic** and **mini-batch** gradient descent can produce **noisy, uneven error curves**, making it **tough to spot** the true minimum validation error.

- **Don't stop** at the **first increase** in validation error. Keep training to see if the error consistently stays above the minimum.
- If the validation error **doesn't improve** after more epochs, it likely indicates **overfitting**.
- Once you're sure, **roll back** to the model parameters where the validation error was lowest.

Stochastic and mini-batch gradient descent

- partial_fit() instead of fit(), to perform incremental learning

```
for epoch in range(n_epochs):
    sgd_reg.partial_fit(X_train_prep, y_train)
    y_valid_predict = sgd_reg.predict(X_valid_prep)
    val_error = mean_squared_error(y_valid, y_valid_predict, squared=False)
    if val_error < best_valid_rmse:
        best_valid_rmse = val_error
        best_model = deepcopy(sgd_reg)
```

- At each epoch, it measures the RMSE on the validation set. If it is lower than the lowest RMSE seen so far, it saves a copy of the model in the best_model variable
- copy.deepcopy(), because it copies both the model's hyperparameters and the learned parameters
- sklearn.base.clone() only copies the model's hyperparameters

Model Optimization Techniques

Feature	Regularization	Learning Curves	Early Stopping
Purpose	Prevents model from fitting noise in the training data	Diagnoses model's performance over time and data capacity	Stops training before overfitting begins
Implementation	Modification of the learning algorithm (e.g., adding penalty terms)	Visualize error on training and validation sets over time or data size	Monitored during training to identify performance deterioration
Focus	Adjusts model complexity or constraints	Observational tool for model behavior over time and data volume	Dynamic adjustment of training duration

Model Optimization Techniques

Feature	Regularization	Learning Curves	Early Stopping
Key Parameter	Penalty strength (e.g., λ in Lasso and Ridge)	Number of training epochs or size of the training dataset	Epoch number at minimal validation error
Outcome	Model with controlled complexity, less prone to overfit	Insight into fitting, overfitting, generalization capabilities	Model that avoids overfitting by halting at an optimal time
Common Use	Part of model formulation in ML contexts	Evaluating and tuning model behavior during training or with more data	Common where resources or time are limited
Dependency on Validation Data	Less direct, focuses on model structure	Highly dependent, shows both training and validation performance	Directly dependent, uses validation error for decisions

Logistic regression

- Logistic regression is a statistical method used for binary classification.

- Predict the probability that an instance belongs to a particular class, typically referred to as the positive class (1) or the negative class (0).

  - whether an email is spam (1) or not spam (0).

- Next Lecture