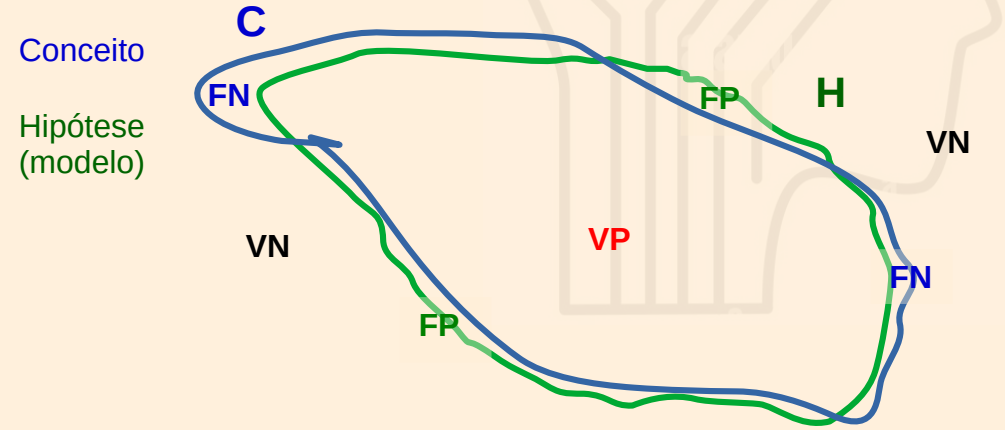
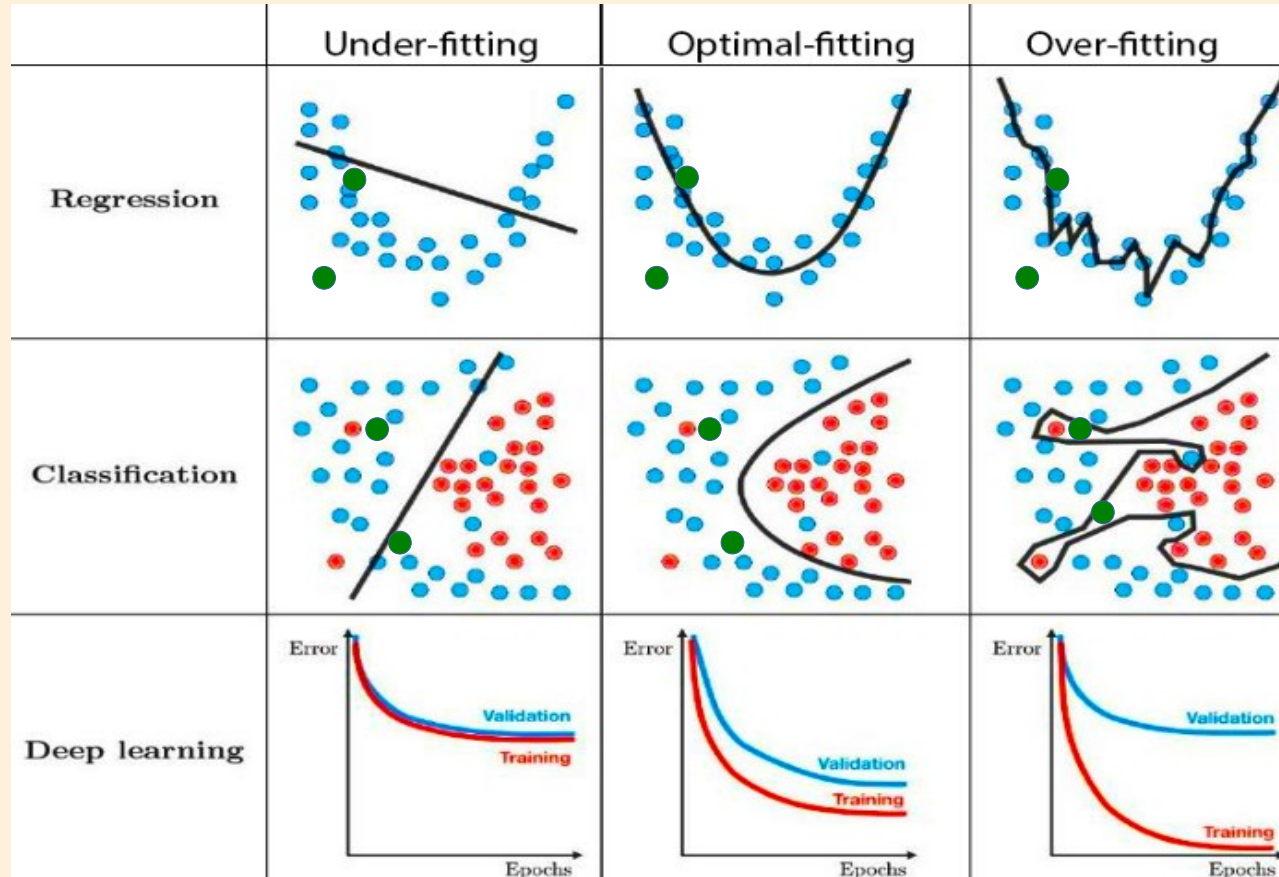


O que faz um bom modelo aprendiz?

- Um bom modelo é preciso
 - Maior nível de acurácia (e precisão) possível
- Generaliza bem
 - Produz mais acertos para novos dados
- Não se superajusta
 - Menos erros de classificação
 - Menos suscetível a anomalias (*outliers*)



Ajustes do modelo

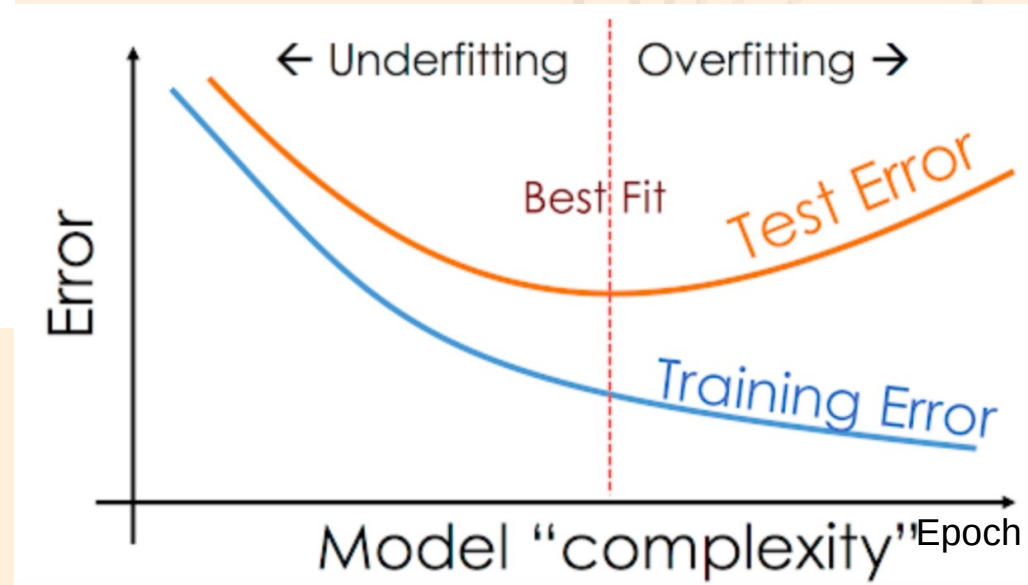
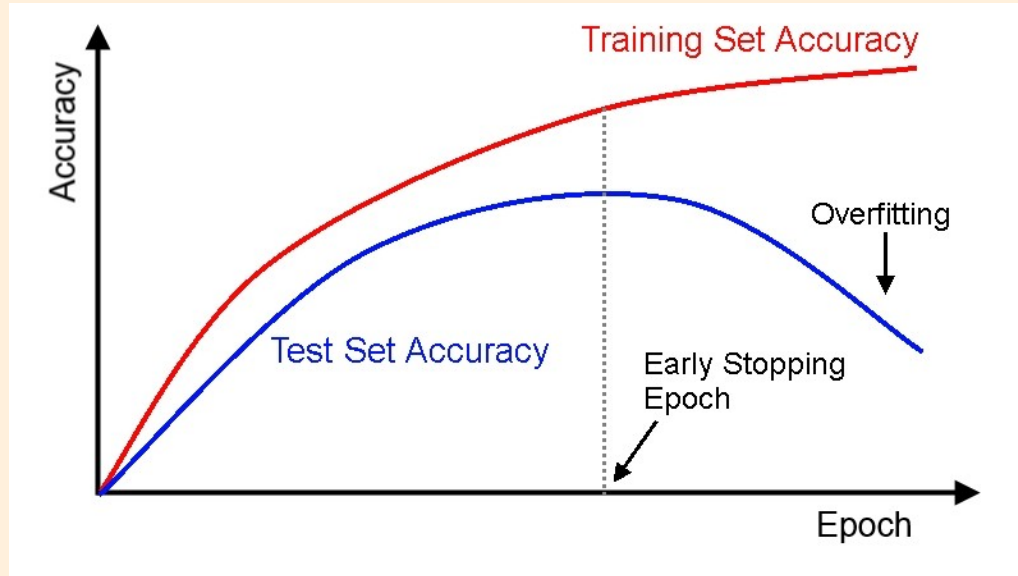


- Treinam – classe A
- Treinam – classe B
- Teste / “novo” dado

Superajuste (overfitting)
ex.

- 95% de acerto para os dados de treinamento
- 70% de acerto para os dados de teste

Acurácia e erro



Regularização



- É um método para tornar o modelo mais geral ao conjunto de dados
- Modos de regularizar
 - L1 & L2 Regularization
 - Cross Validation
 - Early Stopping
 - Drop Out
 - Dataset Augmentation

Modos...

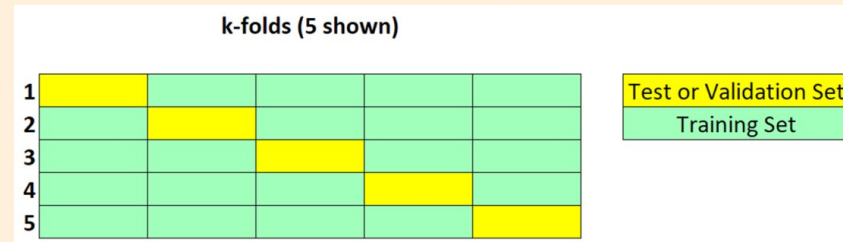
- L1 & L2 *Regularization*
 - Penalizar pesos grandes durante o *backpropagation*
 - λ grau de penalidade
 - L1 – regressão Lasso

$$\text{Erro} = \frac{1}{2}(\text{previsto} - \text{real})^2 + \frac{\lambda}{2} \sum |(w_i)|$$

- L2 – regressão Ridge

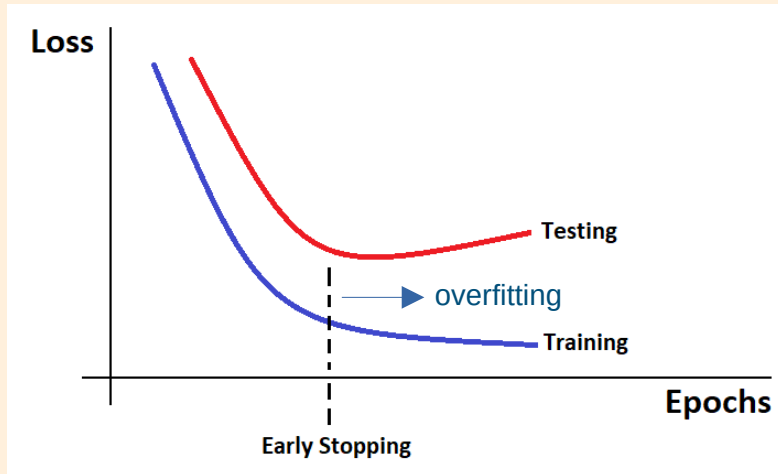
$$\text{Erro} = \frac{1}{2}(\text{previsto} - \text{real})^2 + \frac{\lambda}{2} \sum (w_i)^2$$

- Validação cruzada (*cross validation*)
- Treinamento que divide o conjunto de dados em k partes,
 - Sendo uma parte para teste e as demais para treinamento
 - Revezando as partes e calculando o erro e acurácia médios após este revezamento

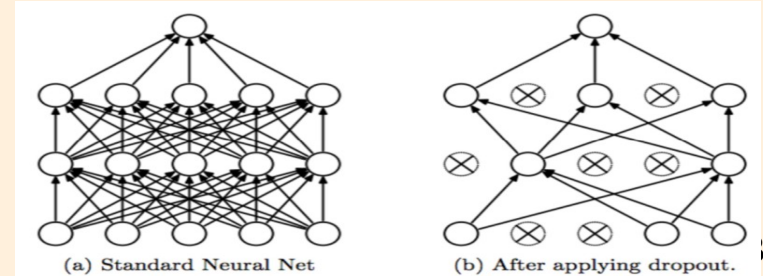


Modos ...

- Early stopping (parada antecipada)
 - Evita o sobreajuste
 - Quando a acurácia ou erro não progridem mais durante n épocas



- Dropout
 - eliminação de nós (ocultos e visíveis) em uma rede neural, **aleatoriamente**
 - com o objetivo de reduzir o sobreajuste.
 - No treinamento, certas partes da rede neural são ignoradas **aleatoriamente** durante o treinamento.
 - Ajuda a reduzir o aprendizado que seja interdependente entre os neurônios.
 - Assim, a RNA aprende as características dos dados mais robustas ou significativas.



Data augmentation

- A ampliação de dados para aprimorar o modelo.
- No conjunto de dados de entrada fazer pequenas variações nele para melhorar a quantidade de dados para treinamento.
- Isso permite criar modelos mais robustos que não se sobreajustam.
- Usado principalmente em visão computacional
- Mas também pode ajudar a incrementar a base de dados com mais variabilidade
 - Porém, com dados plausíveis



Dia	Aspecto	Temper	Umid	Vento	Joga?
D15	nublado	amena	normal	fraco	sim
D16	nublado	fresco	elevada	fraco	sim
D17	sol	fresco	elevada	forte	não
D18	chuva	quente	elevada	fraco	não

Épocas, iterações e lotes

- Uma **época** ocorre quando o conjunto completo dos dados de treinamento é transmitido / propagado para frente e retropropagado em nossa rede neural
- O tamanho do **lote** é o número de amostras de treinamento propagado para frente e retropropagado
- Exemplo: 1000 itens em nosso conjunto de dados,
 - Com um tamanho de lote de 100 registros.
- Portanto, são 10 **iterações** (100 x 10) para concluir uma época.

Base Wine Quality – 80% treinamento = 1279 registros

Treinamento: `mlp.fit(X_train, y_train, epochs=50, batch_size=32, ...)` $1279 / 32 = 39,97 \rightarrow 40$ registros por época

Epoch 1/50

40/40 ----- 1s 7ms/step - accuracy: 0.2987 - loss: 2.1830 - val_accuracy: 0.5281 - val_loss: 1.6385

Epoch 2/50

40/40 ----- 0s 3ms/step - accuracy: 0.5500 - loss: 1.5365 - val_accuracy: 0.5437 - val_loss: 1.3019

Melhores práticas



- Funções de ativação:
 1. ReLU ou Leaky ReLU $\rightarrow [0 \ 1]$
 2. Logistic (sigmoid) $\rightarrow [0 \ 1]$
 3. Softmax (sigmoid multiclasse)
 4. Tangente hiperbólica (tanH) $\rightarrow [-1 \ 1]$
- Função de perda (loss):
 1. MSE para regressão
 2. Binary crossEntropy se saída binária (S/N, V/F, 0/1)
 3. Categorical crossEntropy \rightarrow várias classes, OneHot Encoder
 4. Sparse Categorical Crossentropy \rightarrow não OneHot, classes strings
 - Usar ativação softmax
- Regularização:
 - Use L2 e comece pequeno e aumente gradativamente (0,01, 0,02, 0.05, 0.1.....)
- Dropout entre 0,2 e 0,5
- Taxa de aprendizado: 0,001
- Número de camadas ocultas: tão profundo quanto o desempenho de sua máquina permitir
- Número de épocas: 50 a 500
- Interessante usar Early Stopping