Selección de Modelos

Criterios para Escoger un Modelo

- Error
- Costo computacional de entrenamiento
- Costo computacional en producción
- Habilidad de explicar predicciones

Diagnóstico de problemas con el Modelo

- Lo que buscamos es un modelo con buena generalización
  - Que prediga los valores correctos para ejemplos nuevos
- Si es menos complejo que la función que en realidad genero los datos será incapaz de representarla bien y tendremos "underfitting" o bajo-ajuste
  - Una línea para representar un polinomio de mayor orden
- Si es más complejo tenemos "overfitting" o sobre-ajuste
  - Un polinomio de grado seis para aproximar uno de grado tres
  - Un modelo demasiado complejo aprende también el ruido
- La complejidad del modelo depende de
  - Sus grados de libertad
  - La cantidad de ejemplos que usa para entrenarse

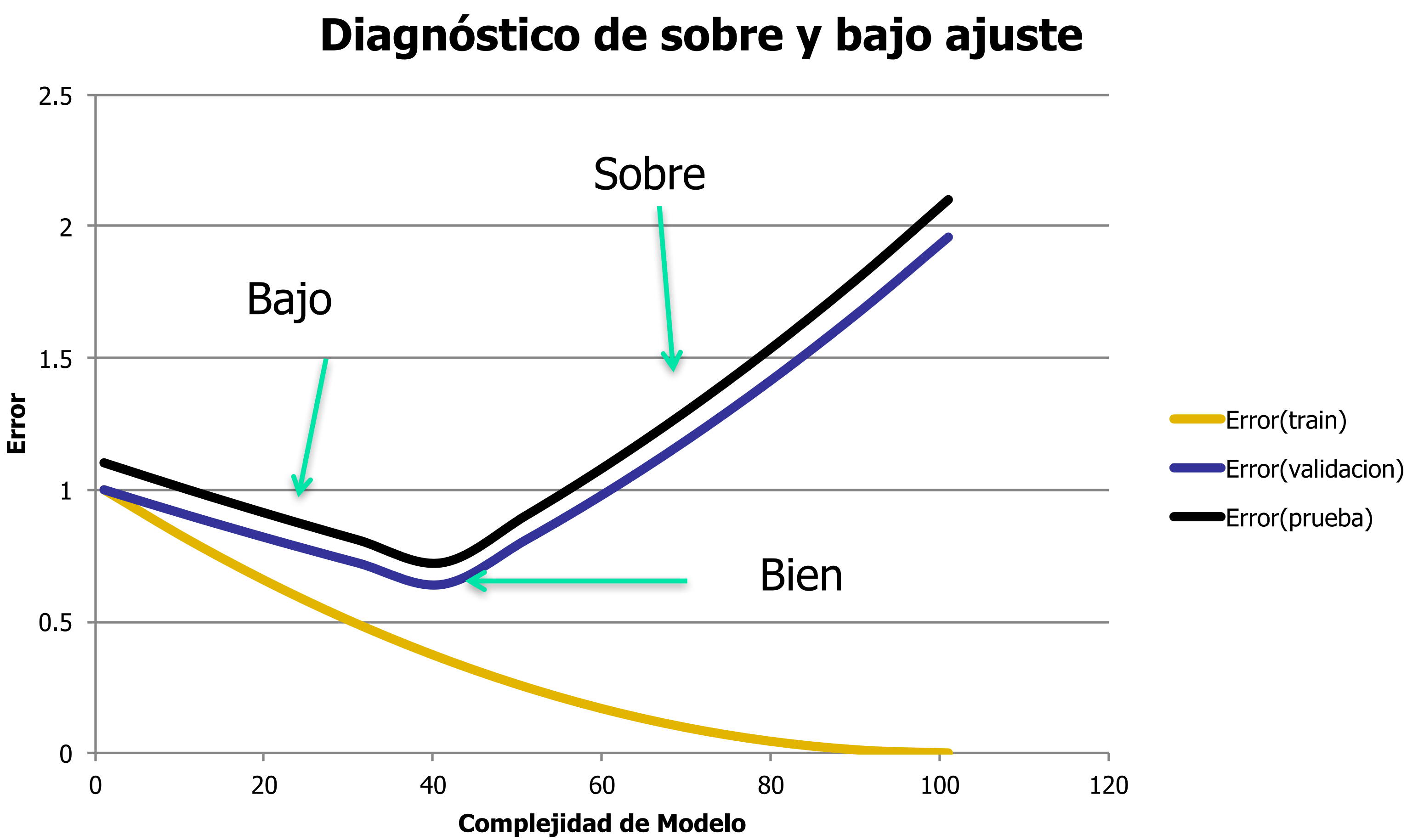
Sobre y bajo ajuste Manifestación

- Bajo-ajuste (Underfitting)
  - Alto error en el conjunto de entrenamiento (Alto sesgo)
- Sobre-ajuste (Overfitting)
  - Bajo error en el conjunto de entrenamiento
  - Alto error en el conjunto de validación y prueba (alta varianza)

¿Cómo determinar la complejidad adecuada?

- Evaluar los errores de entrenamiento y validación con respecto a la complejidad del modelo
  - Para cada valor de complejidad estimar el error de generalización
    - Validación cruzada o bootstrapping

La Complejidad Adecuada



Nota

- Lo anterior nos ayuda a determinar la complejidad apropiada con respecto a un volumen de datos dado
- En ocasiones, sin embargo, es posible mejorar el desempeño del modelo utilizando más datos. En ocasiones no. Dependiendo de la expresividad del modelo

¿Qué hacer?

- Sobre-ajuste (Alta Varianza)
  - Utilizar más ejemplos para entrenar
  - Reducir la complejidad del modelo
    - Reducir el número de atributos (identificar cuales equivalen a ruido)
    - Usar un modelo más simple
      - Reducir el número de neuronas en capas ocultas
      - Aumentar la constante de regularización

¿Qué hacer?

- Bajo Ajuste (Alto sesgo)
  - Agregar atributos
    - Variables derivadas (x*x, x*y, sin(x) etc)
    - Técnicas para transformar variables en algo más fácil de procesar por el modelo (PCA, Factor análisis,....)
    - Nuevas variables acerca del problema
  - Cambiar a un modelo más complejo
    - Decrementar la constante de regularización Lambda
    - Aumentar el número de neuronas en capas ocultas
    - Cambiar de modelo lineal a uno no lineal…
  - Cambiar la distribución de los datos

Estimación del Error y Selección de Híper-parámetros

Elaboración de un Modelo
Ciclo Básico

- Dividir los datos en dos grandes grupos
  - Datos de entrenamiento
    - Datos de aprendizaje y datos de validación
    - Ciclo aprendizaje-validación para escoger los parámetros del modelo ,e.g. Lambda, número de neuronas,…
  - Datos de prueba
  - Los datos de prueba no deben utilizarse en ningún punto del aprendizaje y validación
    - Los datos de prueba deben seleccionarse aleatoriamente a partir de los datos disponibles de manera que representen la distribución original (cuidar que respeten relaciones temporales)
    - Por lo general se apartan entre un 25% y un 33% de los datos para prueba
- Entrenar con todos los datos de entrenamiento
- Reportar el error promedio de validación para los híper-parámetros seleccionados o el de los datos de prueba

Selección de Modelos

- Normalmente las técnicas de aprendizaje cuentan con híper parámetros que determinan su complejidad
  - Lambda, la topología de la red neuronal, etc.
- Los híper parámetros son parámetros del modelo final que hay que aprender
  - El algoritmo para aprenderlos es prueba y error
    - Para cada valor h a explorar del híper parámetro:
      - Entrenar el modelo usando h y calcular su error
    - Seleccionar la h con mejor desempeño
- No debemos usar el conjunto de prueba ni el mismo conjunto de validación para calcular el error de cada h, pues sobre ajustaríamos este aprendizaje
  - Necesitamos un ciclo extra para seleccionarlo: ciclo de aprendizaje-validación

Ciclo aprendizaje-validación

- Lo apropiado es ejecutar el ciclo aprendizaje-validación múltiples veces y obtener el error de aprendizaje promedio
  - Utilizar diferentes semillas para el generador de aleatoriedad
  - Utilizar diferentes datos (o utilizar de manera diferente los datos de entrenamiento)
- ¿Cómo generamos los datos para diferentes corridas del ciclo?
  - Validación cruzada
  - Boostrapping

Ciclo aprendizaje-validación

- Validación cruzada (Cross-validation)
  - Dividimos los datos en k subconjuntos (normalmente 5 o 10)
  - Escogemos uno como validación y los restantes k-1 como aprendizaje
  - Repetimos para cada uno de los k conjuntos
  - Sacamos estadísticas de los k resultados
- Bootstrapping
  - Si tenemos n datos en el conjunto de entrenamiento
  - Sacamos k muestras con reemplazo de tamaño n
  - Dividimos las muestras en aprendizaje y validación
  - Eliminar datos de validación que aparecen en aprendizaje
  - Sacamos estadísticas de los k resultados
- Repetimos el proceso para cada híper parámetro y elegimos el que entregue mejores resultados

Notas

- La selección de variables, normalización, etc., deben ir dentro del ciclo

Ejercicio

- Implement cross-validation to select the appropriate value of lambda for the iterative lineal regression model
  - Use the regLinPoli2.csv data file
  - You can use, if you wish, the Regularization4class.ipynb starter code

Descomposición del Error

El Error

- El error de aprendizaje puede dividirse en tres componentes
  - El error irreductible dado a ruido
  - El error debido al sesgo del modelo. Lo que el modelo no puede capturar de la realidad
  - El error dado a alta varianza. Lo que el modelo captura pero no es real, es solo accidental en los datos de entrenamiento
- El alto sesgo se manifiesta como bajo-ajuste y la alta varianza como sobre-ajuste

Descomposición del Error en Sesgo y Varianza

- Supongamos que la función real es de la forma:
  - $y=f(x) + \varepsilon$, de ε es el ruido que se distribuye normalmente con media cero y varianza σ^2
- Nuestro modelo produce una predicción
  - $V\hat{}(x)$, para toda x
- Medimos el error como
  - $\Sigma(y-V\hat{}\,(x))^{2}$, en el caso de regresión (o de clasificación probabilística)

Descomposición del Error en Sesgo y Varianza

- Queremos estimar el error esperado para un nuevo punto x*

$$Err(x^*)=E[(y-\hat{V}(x^*))^2]$$
$$=E[(f(x^*)+\varepsilon-\hat{V}(x^*))^2]$$
$$=\sigma^2+[E(\hat{V}(x^*))-f(x^*)]^2+E[\hat{V}(x^*)-E(\hat{V}(x^*))]^2$$
$$=\sigma^2+Bias^2(\hat{V}(x^*))+Var(\hat{V}(x^*))$$
$$=ErrorIrreductible+Sesgo^2+Varianza$$

- Normalmente hay un compromiso entre sesgo y varianza

Descomposición del Error en Sesgo y Varianza

- Nótese que estas esperanzas son sobre todo lo que es aleatorio
  - Pesos iniciales (w's iniciales)
  - El conjunto de datos de entrenamiento (es la esperanza estimada sobre todos los posibles conjuntos de entrenamiento)
    - Por ejemplo para una regresion lineal: E(V^(x*)) es la salida esperada del modelo sobre todos los posibles conjuntos de entrenamiento con todas las posibles w's iniciales

Derivación Descomposición de Error

Derivación Versión 1

- Una propiedad importante (truco para derivar)
  - $Var(X)=E(X^2)-[E(X)]^2$
- Sustituimos la variable aleatoria X por la discrepancia de nuestro modelo
  - $Var(V\hat{}(x)-f(x)-\varepsilon)=Var(V\hat{}(x)) + \sigma^2$
    - Porque la varianza de f(x) es cero pues no es una variable aleatoria y la covarianza entre el ruido y V^(x) es cero

Derivación Versión 1

- De la fórmula de la varianza sustituyendo:
- $Var(V\hat{}(x)) + \sigma^2 = E[(V\hat{}(x)-f(x)-\varepsilon)^2]-(E[V\hat{}(x)-f(x)-\varepsilon])^2$
  - $E[(V\hat{}(x)-f(x)-\varepsilon)^2]=MSE$ (error cuadrático medio)
  - $(E[V\hat{}(x)-f(x)-\varepsilon])^2=(E(V\hat{}(x))-E(f(x))-E(\varepsilon))^2$

  $=[E(V\hat{}(x))-f(x)]^2 = Bias^2$
    - Porque $E(f(x))= f(x)$ y $E(\varepsilon) =0$
- Sustituyendo en la primer formula
- $Var(V\hat{}(x)) + \sigma^2=MSE - Bias^2$
- $MSE=Var(V\hat{}(x)) + \sigma^2 + Bias^2$

Version 2
Derivación

- Algunas propiedades importantes:
  1. $E(E(x))=E(x)$
  2. $E((x-E(x))^2)=E[x^2-2xE(x)-E(x)^2]$

     $=E(x^2)-2E(xE(x))+E[E(x)^2]$

     $=E(x^2)-2E(x)E(x)+E(x)^2$

     $=E(x^2)-E(x)^2$
  3. $E(x^2)=E((x-E(x))^2) +E(x)^2$ (fórmula varianza)
  4. $E((c+N(0,\sigma))x)$

     $=E(cx+xN(0,\sigma))=cE(x)$ (la covarianza es cero)

Derivación

- Regresando al error esperado:

$E[(y-V^\wedge(x^*))^2] = E[y^2-2yV^\wedge(x^*)+V^\wedge(x^*)^2]$

$= E[y^2] - 2E(yV^\wedge(x^*)) + E[V^\wedge(x^*)^2]$

$= E((y-E(y))^2) + E(y)^2$ (propiedad 3)

$-2E(V^\wedge(x^*))f(x^*)$ (propiedad 4)

$+ E[(V^\wedge(x^*)-E(V^\wedge(x^*)))^2] + E(V^\wedge(x^*))^2$ (propiedad 3)

$= E((y-E(y))^2)$ (ruido. El desarrollo da σ^2 usando prop.4 y 1)

$+E(y)^2 - 2E(V^\wedge(x^*))f(x^*) + E(V^\wedge(x^*))^2$ (sesgo^2 esto se reduce a $(y-E(V^\wedge(x^*)))^2$ note que $E(y)=y$)

$+E[(V^\wedge(x^*)-E(V^\wedge(x^*)))^2]$ (varianza)