Segundo Parcial – Introducción al Aprendizaje Automático

Humberto Gerardo Peña Páez.

1. **(20 puntos) Investiga/define lo siguiente:** (no copy-paste, hay que interpretar con nuestras palabras)
    - **Aprendizaje Supervisado.**

Es una técnica utilizada en el área de aprendizaje automático o minería de datos que consiste en crear una función capaz de predecir un valor adecuado para un conjunto de entradas tomando con anterioridad un conjunto de datos de entrenamiento, que justamente estos datos de entrenamiento es lo que lo diferencia del aprendizaje no supervisado. En la función creada hay 2 salidas, ya sea un valor numérico, tomado de la regresión o una etiqueta, tomado de la clasificación.

- **Conjunto de Entrenamiento, validación y prueba.**

Son los 3 conjuntos que se utilizan en el aprendizaje supervisado, donde: el conjunto de entrenamiento es el conjunto de datos que utiliza el modelo para entrenarse y tomar ejemplos para dar el mejor resultado; el conjunto de validación es el conjunto de datos donde se probará el modelo previamente entrenado con el fin de obtener métricas y parámetros para evaluar su desempeño obtenido y cuáles son sus posibilidades de fallos al aplicarlos en el mundo real; y el conjunto de prueba es el conjunto de datos con el que donde el modelo previamente seleccionado y entrenado nos arrojará la información del error real del mismo.

- **Función de pérdida.**

Es una métrica utilizada en el aprendizaje supervisado para conocer que tan bien son modelados los datos con el modelo asignado. No existe una sola función de perdida para todos los algoritmos en los cuales podemos basar nuestro modelo, sin embargo, en general, mientras más grande sea el número, las predicciones obtenidas algoritmo seleccionado estarán más desviadas de los resultados reales.

- **Parámetros e hiperparámetros.**

Son variables importantes para el procesamiento de datos en un modelado de aprendizaje supervisado. Los parámetros, son aquellas variables que son obtenidas o resguardadas internamente en el modelado del conjunto de datos, para que el conjunto pueda trabajar con variables más exactas y así nos pueda dar un resultado más optimo, es decir, son las columnas del junto de datos de entrenamiento y validación que son brindadas al modelo para trabajar con ellos. Así mismo, los hiperparámetros son aquellas variables que estan por encima del conjunto de datos, son dadas o asignadas directamente por la persona al momento de llamar al modelo con el algoritmo seleccionado con el fin de ayudar a controlar el proceso de aprendizaje o determinar el valor de los parámetros al que se necesita llegar, para obtener una predicción mucho más exacta y con una perdida mínima.

- **Regularización**

Es un método utilizado en los algoritmos con el fin de tratar el problema de sobreajuste del modelo. Este método, más bien es una metodología ya que no cuenta con una ecuación a seguir, sino que es

aplicando la metodología y evaluar el porcentaje obtenido respecto al sobreajuste inicial. Muchas veces se utiliza para reducir el tamaño del modelo y así poder llegar a un equilibrio entre sobre pasar la capacidad del aprendizaje o que sea un modelo insuficiente y no se nos pueda brindar una buena métrica de evaluación, también se utiliza regularizando pesos, aplicando restricciones fuera del modelo inicial, pero con el mismo fin de poder detener el sobreajuste.

- **Overfitting y Underfitting.**

Son problemas presentados en los problemas modelados del aprendizaje automático, estos problemas lo que nos brindan son malos resultados en la salida de este. En general, los datos de entrenamiento tienen que encajar con la desviación de los datos reales, por lo cual debemos estar pendientes respecto a estos términos, ya que como su nombre lo indica, el overfitting, se conoce como el sobreajuste, es decir brindar datos de entrenamiento al modelo muy por encima de la desviación media de los datos reales a predecir, por cuál el modelo no está generalizado para todo tipo de sitaciones por la sobrecarga de los mismos y la evaluación del error real será muy alta. En contra parte, se encuentra el underfitting, llamado subajuste, que es prácticamente el mismo concepto, solo que a diferencia aquí no se brindan muchos datos poco generalizados, si no, se brindan pocos datos y por encima también poco generalizados, por lo cual nuestro modelo al querer encajar su aprendizaje para predecir un valor no obtendremos el dato que con la menor perdida.

- **Variance-bias tradeoff.**

Es una propiedad de los modelos para el aprendizaje automático que consiste en compensar el sesgo y la varianza, según lo necesite el modelo previamente estudiado y analizado, reduciendo o incrementando el parámetro según lo parezca para no subajustar o sobreajustar el modelo con la finalidad de obtener el menor error real posible.

2. (30 puntos) Describe en tus palabras como funcionan los siguientes modelos de aprendizaje supervisado (elegir al menos 5 modelos). En la descripción se debe mencionar y explicar parámetros e hiperparámetros (máximo 4) asociados al modelo.

- **Regresión lineal.**

Es un algoritmo utilizado en el área de estadística y en aprendizaje automático que consiste en aproximar la relación existente entre 2 variables de entrada, aunque pueden ser también de muchas más si aplicamos el modelo de regresión con múltiples variables, obteniendo la predicción de la tendencia de los datos.
Los parámetros de este modelo son 2 variables de entrada, una variable independiente y otra dependiente. Los hiperparámetros asociados a este modelo son en general si utilizaremos regresión lineal, regresión con múltiples variables o una regresión LASSO.

- **Regresión logística.**

Es un modelo utilizado en aprendizaje automático, consiste en realizar un análisis de múltiples parámetros con el fin de predecir el resultado de una variable categórica, comúnmente se utiliza la regresión logística binaria, es decir, obtener un resultado de un 1 o 0.

Los parámetros que se utilizan en este modelo son: las n variables que tomará el modelo con el fin de predecir. Los hiperparámetros que nos podemos encontrar son: *Solver*, este parámetro se utiliza o modifica para seleccionar el solucionador que el modelo usará, con el fin de encontrar una mejor solución, por mencionar algunos podemos asignarles: *newton-cg, lbfgs, liblinear, sag, saga. Penalty*, utilizado para asignar al modelo que regularización deberá usarse, podemos mencionar: *none, l1, l2 elasticnet. Max_iter*, es el número de iteraciones que el modelo deberá hacer para obtener la predicción. *L1_ratio*, parámetro utilizado cuando el parámetro penalty es elasticnet, sirve para asignar el radio que l1 usará, por defecto tiene el valor none.

- **KNN.**

Este es un algoritmo de clasificación utilizado para reconocer patrones sin necesidad de un aprendizaje como tal. Este algoritmo consiste en clasificar un dato nuevo en el grupo donde tenga k vecinos más cercanos, es decir tomará la distancia de los elementos de diferentes grupos y donde haya más elementos con menor distancia, será incluido ahí y clasificado en ese grupo. Este algoritmo es computacionalmente costoso ya que debe tener cargados y guardados todos los datos en cada elemento nuevo para lograr clasificarlo.
Los parámetros que este modelo utiliza es en general los datos del modelo para poder asignar las vecindades. Los hiperparámetros que el modelo utiliza, por mencionar algunos son: *n_neighbors,* este parámetro se utiliza para seleccionar el número de grupos o vecindades que el algoritmo deberá utilizar para realizar la predicción. *Weights,* modifica el peso de la función predictora, en este caso esta: *uniform* toma en cuenta peso uniforme al vencindario en general y *distance* que toma en cuenta la distancia de cada punto y no del vencindario en general. *Metric,* toma en cuenta la métrica con la que se medirán los datos. *Algorithm,* es para escoger el algoritmo que se utilizará para realizar la predicción, por mencionar algunos: *ball_tree, kd_tree, brute y auto.*

- **Decision Tree.**
- **Random Forest.**
- **Gradient Boosting.**

Es un método de clasificación y de regresión que consiste en realizar predicciones débiles que suelen ser arboles de decisión e ir tomando los resultados para volver a predecirlos y así ir mejorando paso con paso. Se le llama gradient porque la forma en la que construye el modelo suele ser de forma escalonada. Este se detiene cuando llega al número establecido por el usuario o cuando la mejoría entre predicciones es mínima.
Los parámetros que este modelo toma en general son los datos que se tomarán para entrenar el modelo. Los hiperparámetros que utiliza son: *loss,* que especifica la función de perdida que debe ser optimizada, por ejemplo: *deviance, y exponential. N_estimators,* declara el número de etapas de estimación con las que el algoritmo deberá de parar, el default en este parámetro es 100. *Criterion*, especifica la función con la que se determinará la calidad de la división del algoritmo, por mencionar algunos estan: *friendman_mse, squared_error, mse, mae. Max_depth,* se especifica el número máximo de nodos de profundidad que tomará el algoritmo en los árboles de decisión con los que estimará.

- **XGBoost.**
- **Support Vector Machine.**

Es un método de clasificación que separa los datos en dos clases. Consiste en dados los datos de entramiento el algoritmo los separa en dos clases diferentes mediante hiperplanos y basándose en el nuevo dato, donde tenga más relación es la clase que se le asignará. Muchas veces los datos de entrenamiento no son linealmente separables por lo que hay que recurrir a utilizar otros hiperplanos llamados kernel.

Los parámetros que este modelo utiliza son los datos que se utilizarán para el entrenamiento de este. Los hiperparámetros que podemos encontrar son: *kernel*, especifica el kernel del hiperplano con el que se separarán los datos de entrenamiento, por mencionar algunos: *linear, poly, rbf, sigmoid, precomputed*. *Gamma,* es el coeficiente del kernel del algoritmo, únicamente es necesario cuando el kernel es *rbf*, *poly* o *sigmoid*. *Max_iter,* establece el número máximo de iteraciones que el algoritmo realiza. *C,* es el parametro donde se asigna el rango de valores para la regularización del algoritmo.

- **Redes Neuronales.**

Referencias

Common Loss functions in machine learning | by Ravindra Parmar | Towards Data Science

Aprendizaje supervisado - Wikipedia, la enciclopedia libre

Entrenamiento, validación y test - Análisis y Decisión (analisisydecision.es)

Parameters, Hyperparameters, Machine Learning | Towards Data Science

What is the Difference Between a Parameter and a Hyperparameter? (machinelearningmastery.com)

Parameters, Hyperparameters, Machine Learning | Towards Data Science

¿Qué es el sobreajuste u overfitting y por qué debemos evitarlo? (machinelearningparatodos.com)

Machine Learning Supervisado: Fundamentos de la Regresión Lineal | by Victor Roman | Ciencia y Datos | Medium

Qué es overfitting y underfitting y cómo solucionarlo | Aprende Machine Learning

Bias–variance tradeoff - Wikipedia

Ejemplo Regresión Lineal Python | Aprende Machine Learning

Capítulo 6 Regresión logistica binaria | Aprendizaje supervisado en R (fervilber.github.io)

Tune Hyperparameters for Classification Machine Learning Algorithms (machinelearningmastery.com)

sklearn.svm.SVC — scikit-learn 1.0 documentation

sklearn.neighbors.KNeighborsClassifier — scikit-learn 1.0 documentation

sklearn.ensemble.GradientBoostingClassifier — scikit-learn 1.0 documentation