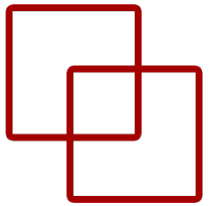
Fundamentos de algoritmos Ensemble

Manuel Castillo-Cara, Luis Sarro
www.manuelcastillo.eu
Departamento de Inteligencia Artificial
Escuela Técnica Superior de Ingeniería Informática
Universidad Nacional de Educación a Distancia (UNED)

Índice



- Background
- ¿Por qué aprendizaje ensemble?
- Cómo trabaja ensemble learning
- Combinar predicciones
- Taxonomía en algoritmos ensemble
- Modelos Múltiples
- Mezcla de expertos (MoE)

Background

Sabiduría de la multitud

¿Hacemos caso a uno o varios médicos?

Sabiduría de la multitud

¿Hacemos caso a uno o varios médicos?

Aprendizaje de conjunto
Estimaciones

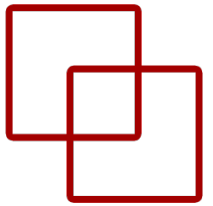
Sabiduría de multitudes

Estimaciones

¿Cómo podemos calcular su peso?

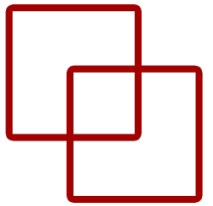
Preguntando a muchas personas y calculando el promedio → se aproximará muchísimo al peso real

Aprendizaje de conjunto



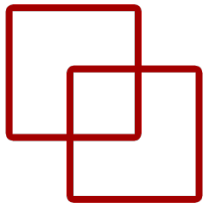
- Según Lior Rokach's 2010 en su libro titulado "*Pattern Classification Using Ensemble Methods*" (pag. 22), describe las decisiones basad en grupos de personas como:
  - **Diversidad de opiniones**: Cada miembro debe tener información privada, aunque sólo sea una interpretación excéntrica de los hechos conocidos.

Aprendizaje de conjunto



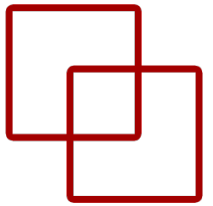
- Según Lior Rokach's 2010 en su libro titulado "*Pattern Classification Using Ensemble Methods*" (pag. 22), describe las decisiones basad en grupos de personas como:
  - **Diversidad de opiniones**: Cada miembro debe tener información privada, aunque sólo sea una interpretación excéntrica de los hechos conocidos.
  - **Independencia**: Las opiniones de los miembros no están determinadas por las opiniones de quienes les rodean.

Aprendizaje de conjunto



- Según Lior Rokach's 2010 en su libro titulado "*Pattern Classification Using Ensemble Methods*" (pag. 22), describe las decisiones basad en grupos de personas como:
  - **Diversidad de opiniones**: Cada miembro debe tener información privada, aunque sólo sea una interpretación excéntrica de los hechos conocidos.
  - **Independencia**: Las opiniones de los miembros no están determinadas por las opiniones de quienes les rodean.
  - **Descentralización**: Los miembros pueden especializarse y sacar conclusiones basadas en el conocimiento local.

Aprendizaje de conjunto



- Según Lior Rokach's 2010 en su libro titulado "*Pattern Classification Using Ensemble Methods*" (pag. 22), describe las decisiones basad en grupos de personas como:
  - **Diversidad de opiniones**: Cada miembro debe tener información privada, aunque sólo sea una interpretación excéntrica de los hechos conocidos.
  - **Independencia**: Las opiniones de los miembros no están determinadas por las opiniones de quienes les rodean.
  - **Descentralización**: Los miembros pueden especializarse y sacar conclusiones basadas en el conocimiento local.
  - **Agregación**: Existe algún mecanismo para convertir los juicios privados en una decisión colectiva.

Aprendizaje de conjunto

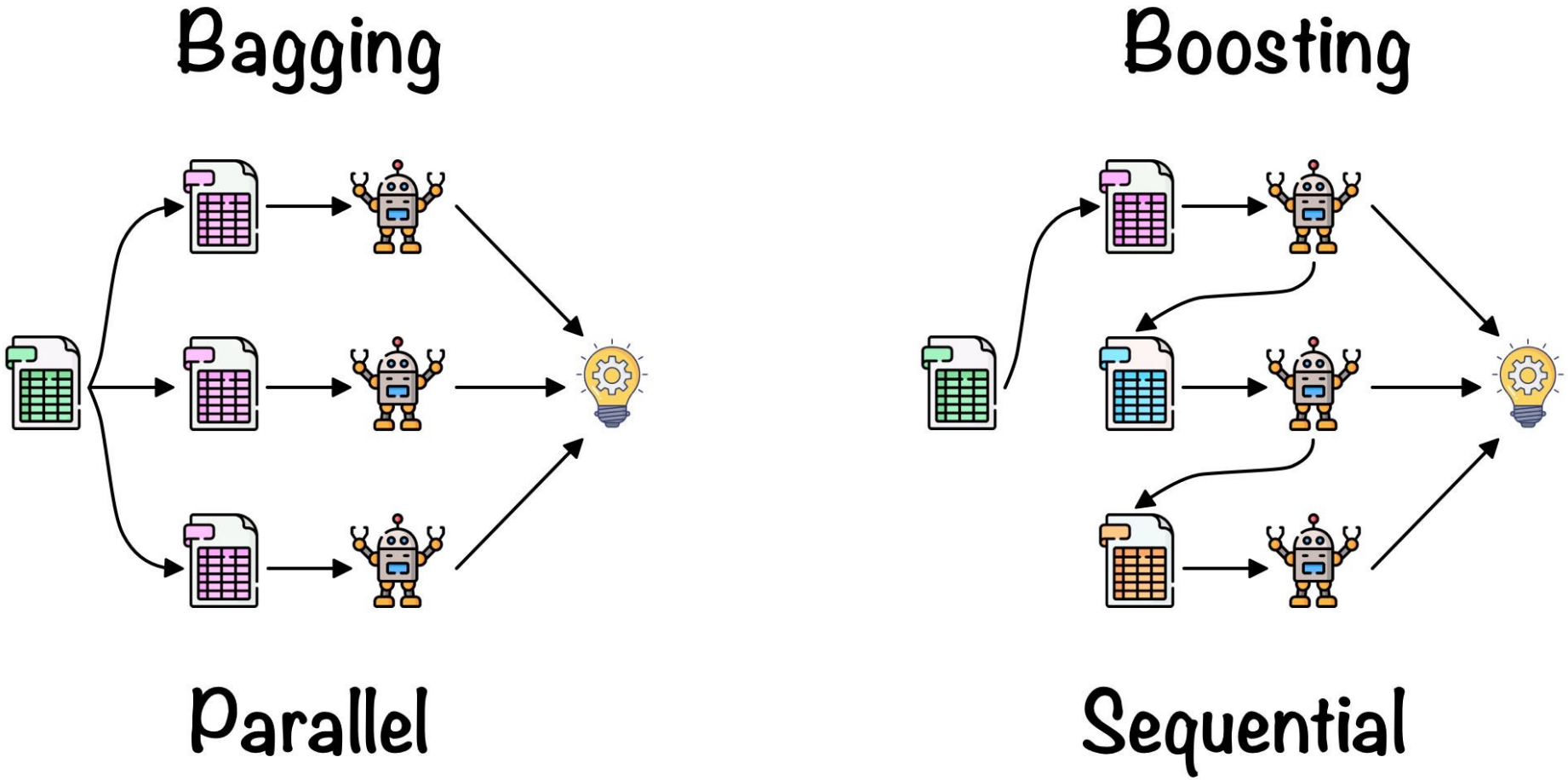
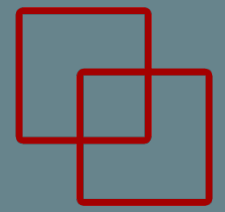


Imagen tomada de: https://towardsdatascience.com/ensemble-learning-bagging-boosting-3098079e5422

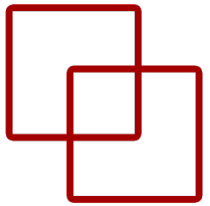


UNED

ETS de Ingeniería Informática

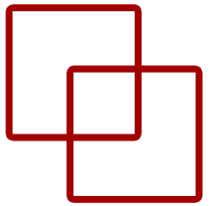
¿Por qué aprendizaje ensemble?

Razones de utilizar ensemble



- Dos razones principales:
  - **Rendimiento:** Puede hacer mejores predicciones y lograr un mejor rendimiento que cualquier modelo unitario.
  - **Robustez**: Ensemble reduce la dispersión de las predicciones y el rendimiento del modelo.

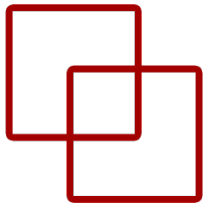
Razones de utilizar ensemble



- Dos razones principales:
  - **Rendimiento:** Puede hacer mejores predicciones y lograr un mejor rendimiento que cualquier modelo unitario.
  - **Robustez**: Ensemble reduce la dispersión de las predicciones y el rendimiento del modelo.

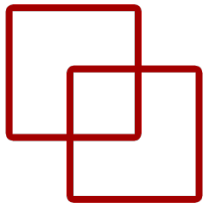
- Esto se entiende como que el modelo reduce el componente de varianza del error de predicción añadiendo sesgo
  - Es decir, en el contexto del **equilibrio entre sesgo y varianza**.

Mejorar la robustez



- Lo más simple: ajustar el modelo varias veces en train y **combinar las predicciones** utilizando una estadística de resumen (e.g., media o moda).

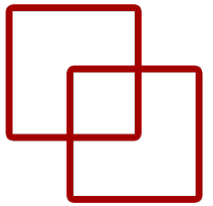
Mejorar la robustez



- Lo más simple: ajustar el modelo varias veces en train y **combinar las predicciones** utilizando una estadística de resumen (e.g., media o moda).

- Importante: cada modelo debe ser ligeramente **diferente**.

Mejorar la robustez

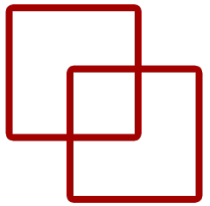


- Lo más simple: ajustar el modelo varias veces en train y **combinar las predicciones** utilizando una estadística de resumen (e.g., media o moda).

- Importante: cada modelo debe ser ligeramente **diferente**.

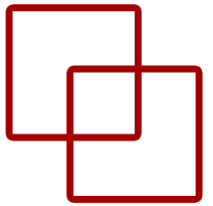
- El rendimiento medio probablemente será aproximadamente el mismo, aunque el rendimiento en el peor y el mejor de los casos se acercará más al rendimiento medio.
  - De hecho, **suaviza** el rendimiento esperado del modelo.

Sesgo y varianza



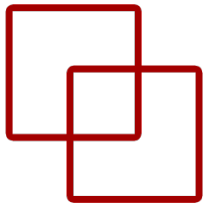
- Los errores cometidos por un modelo se describen en dos propiedades:

Sesgo y varianza



- Los errores cometidos por un modelo se describen en dos propiedades:
  - **Sesgo:** qué tan cerca el modelo puede capturar la función de mapeo entre entradas y salidas.
    - Capta la rigidez del modelo, i.e., la fuerza de la suposición que tiene el modelo sobre la forma funcional del mapeo entre entradas y salidas.
  - **Varianza**: cómo cambia el rendimiento del modelo cuando se ajusta a diferentes datos de entrenamiento
    - Por tanto, se refiere a la cantidad en la que el modelo cambiaría si lo estimamos utilizando un *train* diferente.

Sesgo y varianza



- Los errores cometidos por un modelo se describen en dos propiedades:
  - **Sesgo:** qué tan cerca el modelo puede capturar la función de mapeo entre entradas y salidas.
    - Capta la rigidez del modelo, i.e., la fuerza de la suposición que tiene el modelo sobre la forma funcional del mapeo entre entradas y salidas.
  - **Varianza**: cómo cambia el rendimiento del modelo cuando se ajusta a diferentes datos de entrenamiento
    - Por tanto, se refiere a la cantidad en la que el modelo cambiaría si lo estimamos utilizando un *train* diferente.
- Lo **ideal** es tener un modelo con **bajo sesgo y varianza**, ¡pero es un desafío!
  - A menudo es fácil reducir el sesgo aumentando la varianza.
  - Por el contrario, es fácil reducir la varianza aumentando el sesgo.

Sesgo, varianza, y ensemble

- Ensemble proporciona una manera de **reducir la varianza** de las predicciones
  - Esa es la cantidad de error en las predicciones realizadas que se puede atribuir a la varianza.

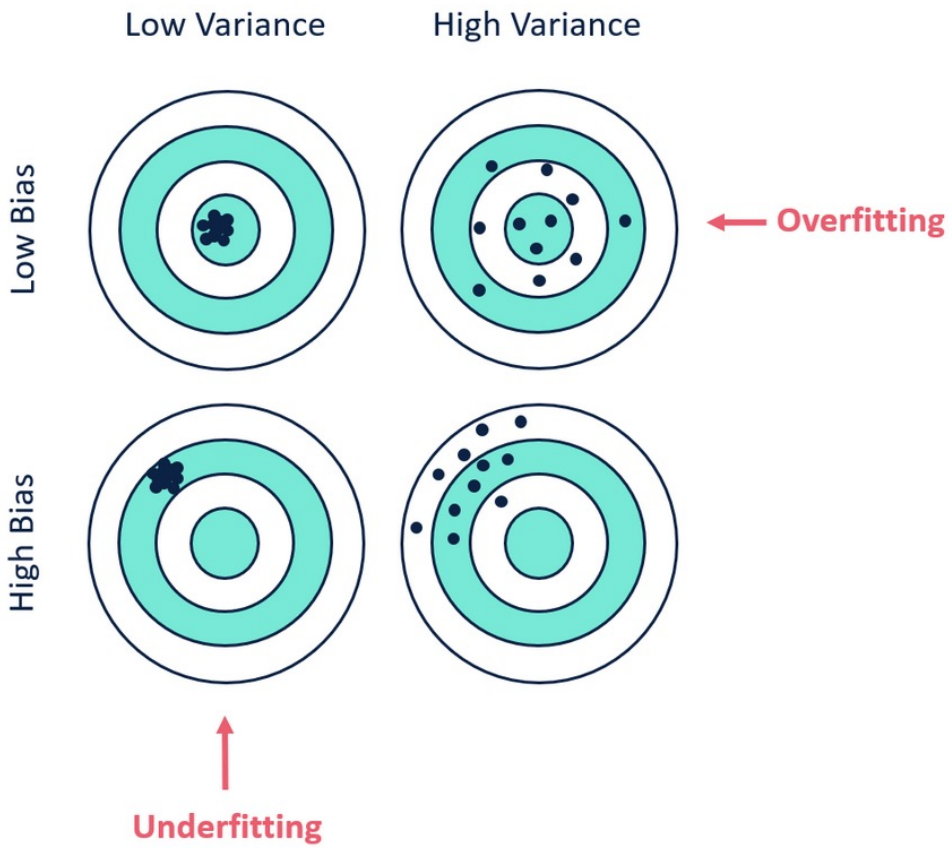


Imagen tomada de: https://medium.com/mlearning-ai/bias-vs-variance-trade-off-d46a410412e0

Sesgo, varianza, y ensemble

- Ensemble proporciona una manera de **reducir la varianza** de las predicciones
  - Esa es la cantidad de error en las predicciones realizadas que se puede atribuir a la varianza.
- Normalmente, una **reducción en la varianza → mejor rendimiento del modelo**.

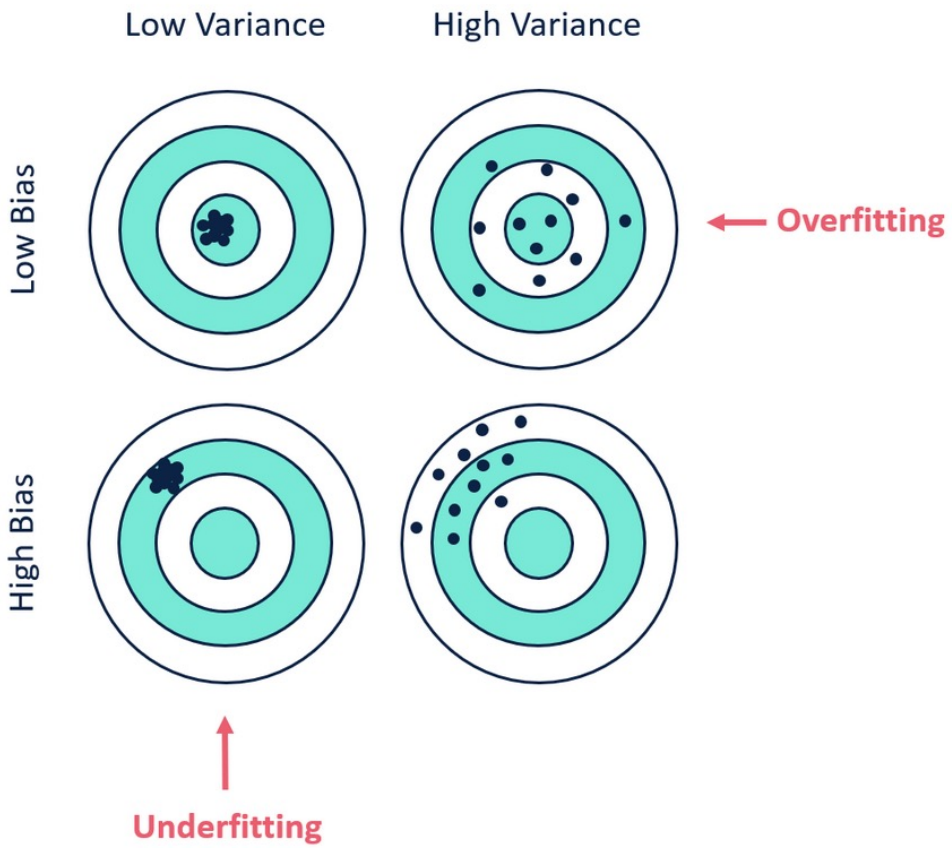


Imagen tomada de: https://medium.com/mlearning-ai/bias-vs-variance-trade-off-d46a410412e0

Sesgo, varianza, y ensemble

- Ensemble proporciona una manera de **reducir la varianza** de las predicciones
  - Esa es la cantidad de error en las predicciones realizadas que se puede atribuir a la varianza.
- Normalmente, una **reducción en la varianza → mejor rendimiento del modelo**.
- Algunas técnicas de ensembles, e.g., Bagging, actúan como un mecanismo de **reducción de la varianza** (del error).

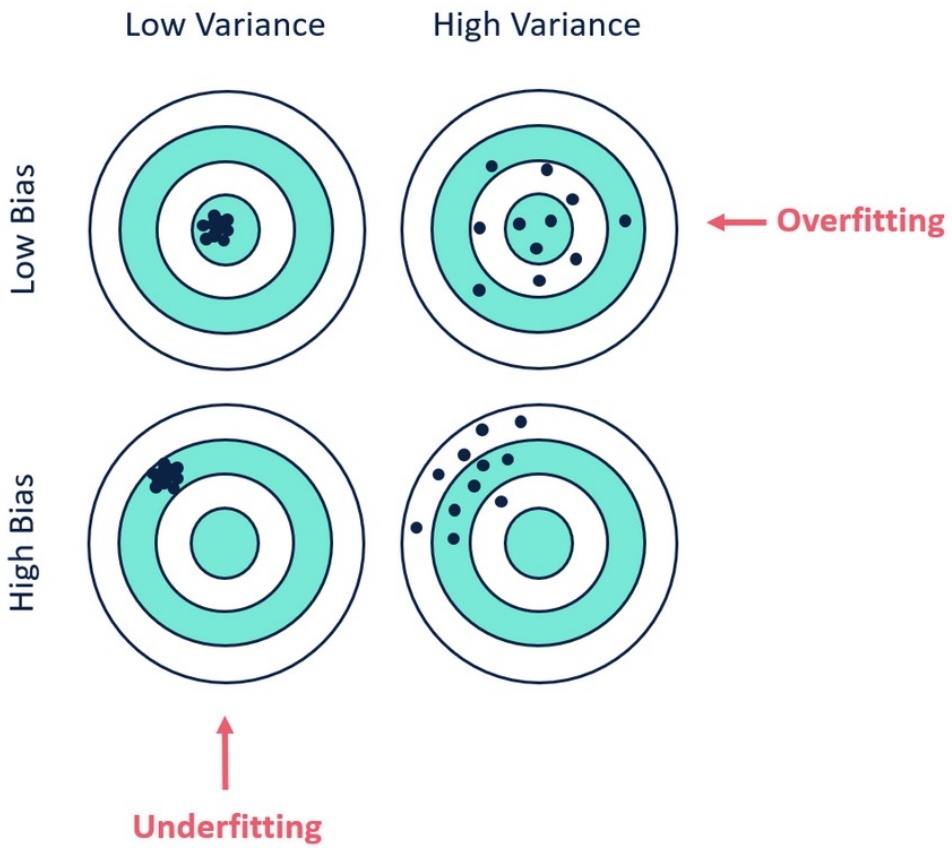


Imagen tomada de: https://medium.com/mlearning-ai/bias-vs-variance-trade-off-d46a410412e0

Sesgo, varianza, y ensemble

- Ensemble proporciona una manera de **reducir la varianza** de las predicciones
  - Esa es la cantidad de error en las predicciones realizadas que se puede atribuir a la varianza.
- Normalmente, una **reducción en la varianza → mejor rendimiento del modelo**.
- Algunas técnicas de ensembles, e.g., Bagging, actúan como un mecanismo de **reducción de la varianza** (del error).
- Otras técnicas de conjunto, e.g., AdaBoost, **reducen** tanto la parte de **sesgo** como la parte de **varianza** del error.

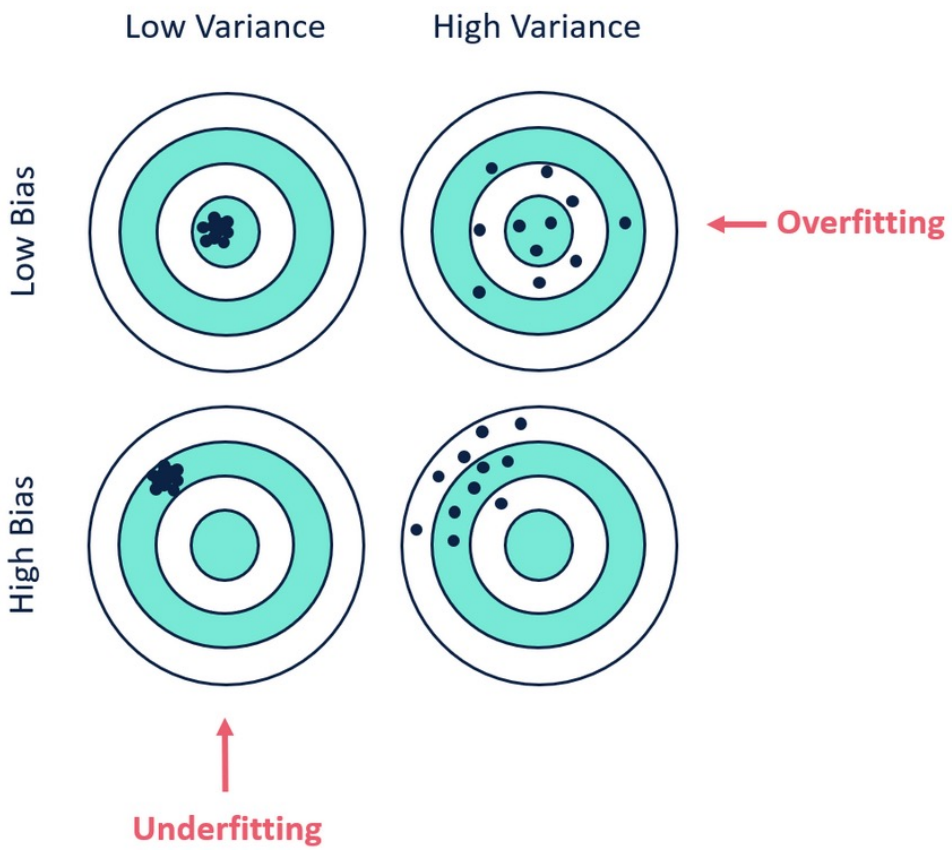
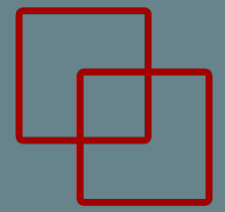


Imagen tomada de: https://medium.com/mlearning-ai/bias-vs-variance-trade-off-d46a410412e0

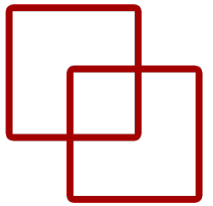


UNED

ETS de Ingeniería Informática

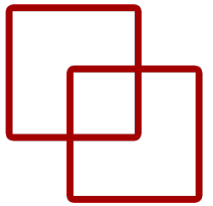
Cómo trabaja ensemble learning

Enfoque principal



- Requieren que los modelos contribuyentes sean **diferentes e independientes** para que cometan errores diferentes y combinar las predicciones:
  - La esperanza es que **promediar sus predicciones mejore su rendimiento**, si cada modelo hace predicciones diferentes.
  - El modelo 1 funcionará mejor y el modelo 2 funcionará peor, y lo contrario para otros casos específicos → Promediar sus predicciones busca reducir estos errores en las predicciones realizadas por ambos modelos.

Enfoque principal



- Requieren que los modelos contribuyentes sean **diferentes e independientes** para que cometan errores diferentes y combinar las predicciones:
  - La esperanza es que **promediar sus predicciones mejore su rendimiento**, si cada modelo hace predicciones diferentes.
  - El modelo 1 funcionará mejor y el modelo 2 funcionará peor, y lo contrario para otros casos específicos → Promediar sus predicciones busca reducir estos errores en las predicciones realizadas por ambos modelos.
- Para que los modelos hagan predicciones diferentes, deben **hacer suposiciones diferentes** sobre el problema de predicción, i.e., una función de mapeo diferente de las entradas a las salidas.
  - Se puede entrenar cada modelo en una muestra diferente en train; o
  - Entrenando distintos tipos de modelos.

Ensemble en clasificación

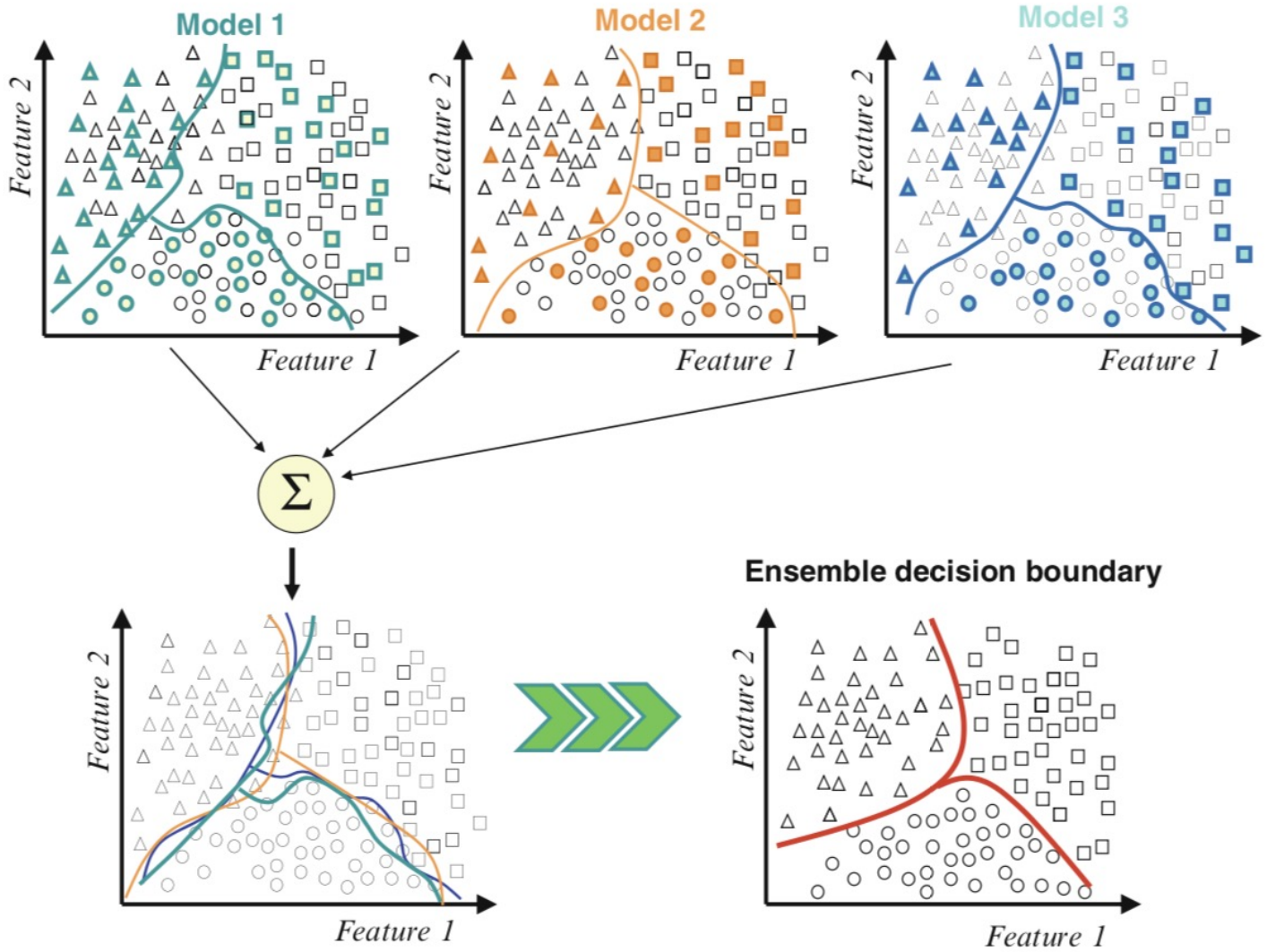


Imagen tomada del libro “Ensemble Machine Learning”, 2012

Ensemble en regresión

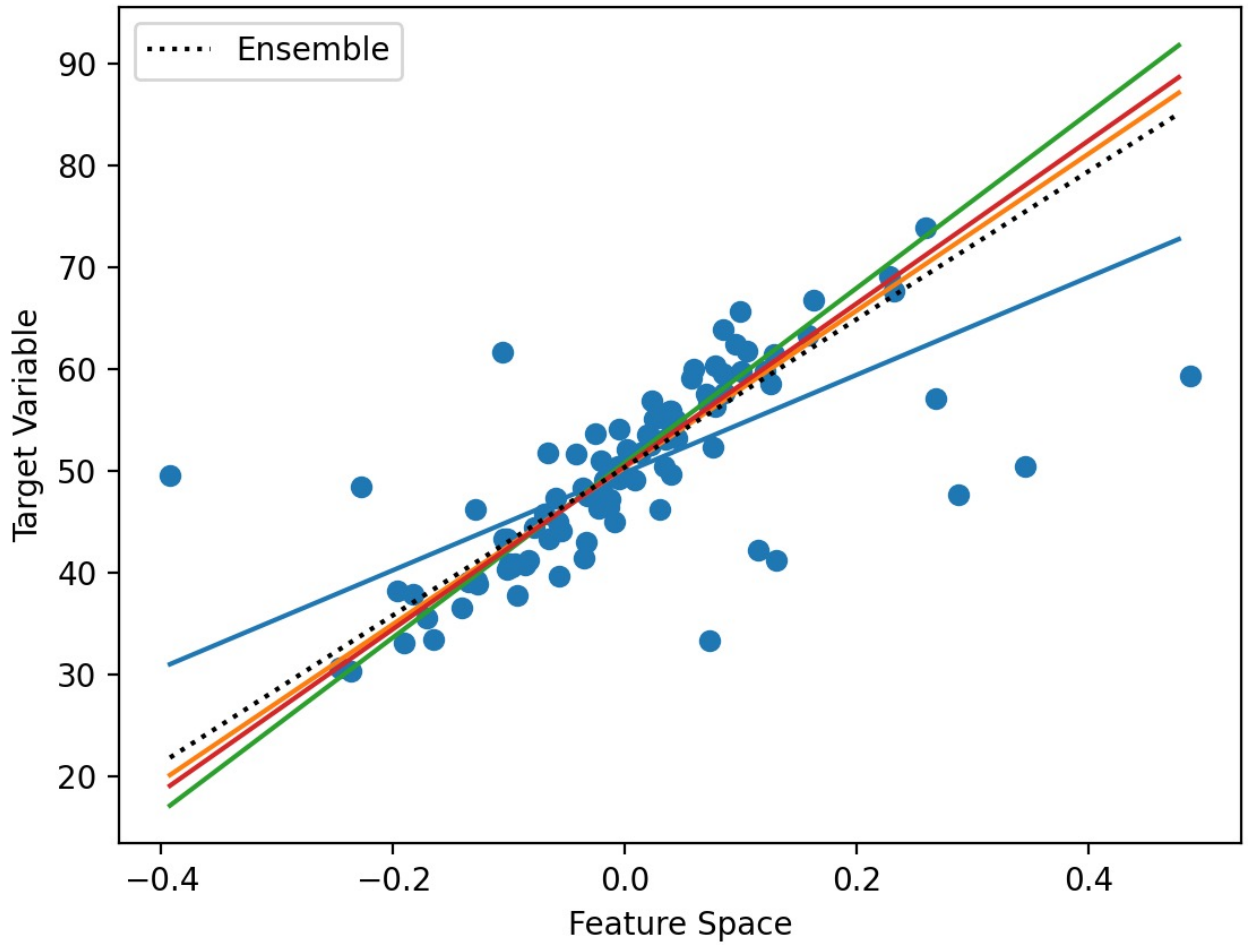
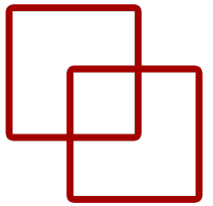


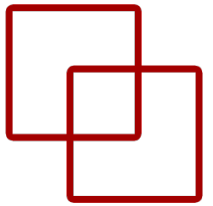
Imagen tomada del libro “Ensemble Machine Learning”, 2012

Diversidad de conjuntos



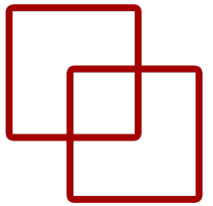
- La diversidad se refiere a las **diferencias** en las decisiones o predicciones hechas por los contribuyentes.

Diversidad de conjuntos



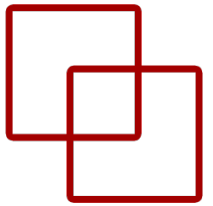
- La diversidad se refiere a las **diferencias** en las decisiones o predicciones hechas por los contribuyentes.
- Dos miembros de un conjunto que hacen predicciones idénticas se consideran **no diversos**.

Diversidad de conjuntos



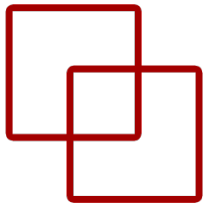
- La diversidad se refiere a las **diferencias** en las decisiones o predicciones hechas por los contribuyentes.
- Dos miembros de un conjunto que hacen predicciones idénticas se consideran **no diversos**.
- Los conjuntos que hacen predicciones completamente diferentes en todos los casos son **máximamente diversos**, aunque esto es muy improbable.

Diversidad de conjuntos



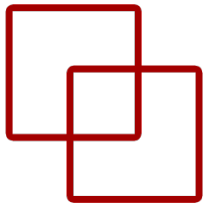
- La diversidad se refiere a las **diferencias** en las decisiones o predicciones hechas por los contribuyentes.
- Dos miembros de un conjunto que hacen predicciones idénticas se consideran **no diversos**.
- Los conjuntos que hacen predicciones completamente diferentes en todos los casos son **máximamente diversos**, aunque esto es muy improbable.
- Se requiere **cierto nivel de diversidad** en las predicciones para construir un buen conjunto.

Diversidad de conjuntos



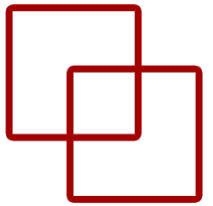
- La diversidad se refiere a las **diferencias** en las decisiones o predicciones hechas por los contribuyentes.
- Dos miembros de un conjunto que hacen predicciones idénticas se consideran **no diversos**.
- Los conjuntos que hacen predicciones completamente diferentes en todos los casos son **máximamente diversos**, aunque esto es muy improbable.
- Se requiere **cierto nivel de diversidad** en las predicciones para construir un buen conjunto.
- Por tanto, la **diversidad** significa que las predicciones por cada contribuyente **son independientes y no están correlacionadas**.
  - *"... if the learners are independent, i.e., [correlation] = 0, the ensemble will achieve a factor of T of error reduction than the individual learners; if the learners are totally correlated, i.e., [correlation] = 1, no gains can be obtained from the combination"*. — Page 99, Ensemble Methods, 2012.

Diversidad de conjuntos
Enfoques para generar diversidad (I)



- Zhi-Hua Zhou en el Cap. 5 del libro titulado "*Ensemble Methods: Foundations and Algorithms*", propone 4 enfoques para generar la diversidad:

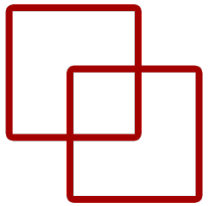
Diversidad de conjuntos
Enfoques para generar diversidad (I)



- Zhi-Hua Zhou en el Cap. 5 del libro titulado "*Ensemble Methods: Foundations and Algorithms*", propone 4 enfoques para generar la diversidad:

  - **Manipulación de muestras de datos**: e.g., muestrear *train* de forma diferente para cada uno de los modelos.

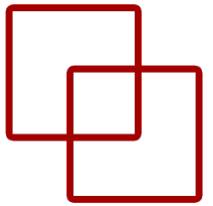
Diversidad de conjuntos
Enfoques para generar diversidad (I)



- Zhi-Hua Zhou en el Cap. 5 del libro titulado "*Ensemble Methods: Foundations and Algorithms*", propone 4 enfoques para generar la diversidad:

  - **Manipulación de muestras de datos**: e.g., muestrear *train* de forma diferente para cada uno de los modelos.
  - **Manipulación de características de entrada**: e.g., entrenar cada modelo en diferentes grupos de características de entrada.

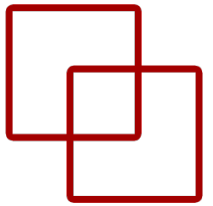
Diversidad de conjuntos
Enfoques para generar diversidad (I)



- Zhi-Hua Zhou en el Cap. 5 del libro titulado "*Ensemble Methods: Foundations and Algorithms*", propone 4 enfoques para generar la diversidad:

  - **Manipulación de muestras de datos**: e.g., muestrear *train* de forma diferente para cada uno de los modelos.
  - **Manipulación de características de entrada**: e.g., entrenar cada modelo en diferentes grupos de características de entrada.
  - **Manipulación de parámetros de aprendizaje**: e.g., entrenar modelos con diferentes valores de hiperparámetros.

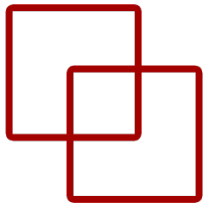
Diversidad de conjuntos
Enfoques para generar diversidad (I)



- Zhi-Hua Zhou en el Cap. 5 del libro titulado "*Ensemble Methods: Foundations and Algorithms*", propone 4 enfoques para generar la diversidad:

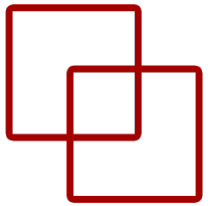
  - **Manipulación de muestras de datos**: e.g., muestrear *train* de forma diferente para cada uno de los modelos.
  - **Manipulación de características de entrada**: e.g., entrenar cada modelo en diferentes grupos de características de entrada.
  - **Manipulación de parámetros de aprendizaje**: e.g., entrenar modelos con diferentes valores de hiperparámetros.
  - **Manipulación de la representación de salida**: e.g., entrenar modelos con valores objetivo modificados de forma diferente.

Diversidad de conjuntos
Enfoques para generar diversidad (II)



- Lior Rokach en el Cap. 4 del libro titulado *"Pattern Classification Using Ensemble Methods"*, propone otra taxonomía similar:

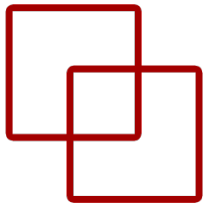
Diversidad de conjuntos
Enfoques para generar diversidad (II)



- Lior Rokach en el Cap. 4 del libro titulado *"Pattern Classification Using Ensemble Methods"*, propone otra taxonomía similar:

  - **Manipular el Inductor:** manipular cómo se entrenan los modelos → Variar los hiperparámetros, Variar el punto de partida, Variar el algoritmo de optimización.

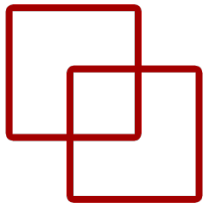
Diversidad de conjuntos
Enfoques para generar diversidad (II)



- Lior Rokach en el Cap. 4 del libro titulado *"Pattern Classification Using Ensemble Methods"*, propone otra taxonomía similar:

  - **Manipular el Inductor:** manipular cómo se entrenan los modelos → Variar los hiperparámetros, Variar el punto de partida, Variar el algoritmo de optimización.
  - **Manipular la muestra de entrenamiento**: manipular los datos utilizados para el entrenamiento → Remuestreo.

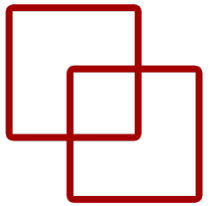
Diversidad de conjuntos
Enfoques para generar diversidad (II)



- Lior Rokach en el Cap. 4 del libro titulado *"Pattern Classification Using Ensemble Methods"*, propone otra taxonomía similar:

  - **Manipular el Inductor:** manipular cómo se entrenan los modelos → Variar los hiperparámetros, Variar el punto de partida, Variar el algoritmo de optimización.
  - **Manipular la muestra de entrenamiento**: manipular los datos utilizados para el entrenamiento → Remuestreo.
  - **Cambiar la representación del atributo objetivo:** manipulando la variable objetivo → Variar codificación, Códigos de corrección de errores, Cambio de etiquetas.

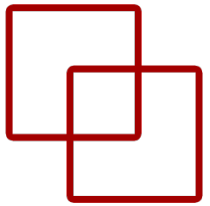
Diversidad de conjuntos
Enfoques para generar diversidad (II)



- Lior Rokach en el Cap. 4 del libro titulado *"Pattern Classification Using Ensemble Methods"*, propone otra taxonomía similar:

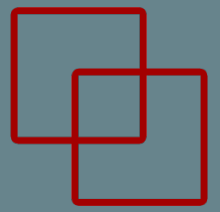
  - **Manipular el Inductor:** manipular cómo se entrenan los modelos → Variar los hiperparámetros, Variar el punto de partida, Variar el algoritmo de optimización.
  - **Manipular la muestra de entrenamiento**: manipular los datos utilizados para el entrenamiento → Remuestreo.
  - **Cambiar la representación del atributo objetivo:** manipulando la variable objetivo → Variar codificación, Códigos de corrección de errores, Cambio de etiquetas.
  - **Particionar el espacio de búsqueda**: manipular el número de características de entrada → Subespacio aleatorio, Selección de características.

Diversidad de conjuntos
Enfoques para generar diversidad (II)



- Lior Rokach en el Cap. 4 del libro titulado *"Pattern Classification Using Ensemble Methods"*, propone otra taxonomía similar:

  - **Manipular el Inductor:** manipular cómo se entrenan los modelos → Variar los hiperparámetros, Variar el punto de partida, Variar el algoritmo de optimización.
  - **Manipular la muestra de entrenamiento**: manipular los datos utilizados para el entrenamiento → Remuestreo.
  - **Cambiar la representación del atributo objetivo:** manipulando la variable objetivo → Variar codificación, Códigos de corrección de errores, Cambio de etiquetas.
  - **Particionar el espacio de búsqueda**: manipular el número de características de entrada → Subespacio aleatorio, Selección de características.
  - **Hibridación**: distintos tipos de modelos o una combinación de los métodos anteriores.

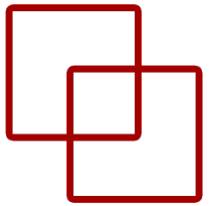


UNED

ETS de Ingeniería Informática

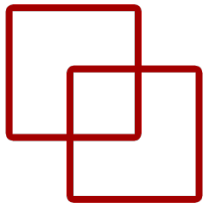
Combinar predicciones

Combinación para clasificación



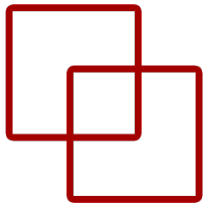
- La clasificación se refiere a problemas de modelado predictivo que implican **predecir una etiqueta de clase** dada una entrada.

Combinación para clasificación



- La clasificación se refiere a problemas de modelado predictivo que implican **predecir una etiqueta de clase** dada una entrada.
- Puede ser una etiqueta de **clase nítida** o puede ser una **probabilidad** de pertenencia a una clase.

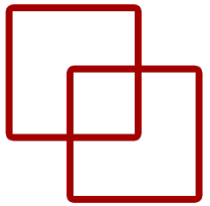
Combinación para clasificación



- La clasificación se refiere a problemas de modelado predictivo que implican **predecir una etiqueta de clase** dada una entrada.
- Puede ser una etiqueta de **clase nítida** o puede ser una **probabilidad** de pertenencia a una clase.
- En el caso de evaluar **probabilidades** predichas:
  - Se pueden convertir en etiquetas de clase nítidas seleccionando un umbral de corte; o
  - Evaluarse utilizando métricas especializadas como la entropía cruzada.

Combinación para clasificación

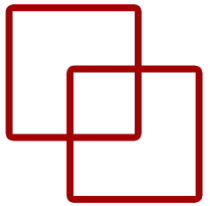
Votación por etiqueta de clase



- La votación implica que cada modelo que hace una predicción **asigna un voto** a esa clase.

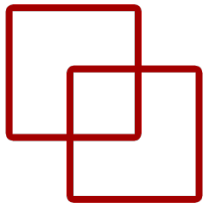
Combinación para clasificación

Votación por etiqueta de clase



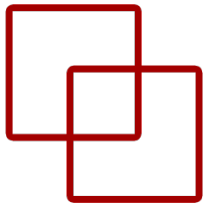
- La votación implica que cada modelo que hace una predicción **asigna un voto** a esa clase.
- Un resultado se escoge utilizando los votos o el conteo.

Combinación para clasificación
Votación por etiqueta de clase



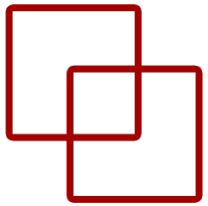
- La votación implica que cada modelo que hace una predicción **asigna un voto** a esa clase.
- Un resultado se escoge utilizando los votos o el conteo.
- Hay muchos tipos de votación, los cuatro votos más comunes son:
  - **Pluralidad** → selecciona la etiqueta de clase con más votos (**moda** o valor más común).

Combinación para clasificación
Votación por etiqueta de clase



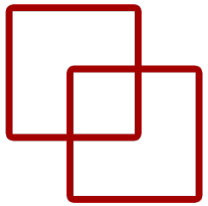
- La votación implica que cada modelo que hace una predicción **asigna un voto** a esa clase.
- Un resultado se escoge utilizando los votos o el conteo.
- Hay muchos tipos de votación, los cuatro votos más comunes son:
  - **Pluralidad** → selecciona la etiqueta de clase con más votos (**moda** o valor más común).
  - **Mayoritario** → requiere la mitad de los votos (una **mayoría**).

Combinación para clasificación
Votación por etiqueta de clase



- La votación implica que cada modelo que hace una predicción **asigna un voto** a esa clase.
- Un resultado se escoge utilizando los votos o el conteo.
- Hay muchos tipos de votación, los cuatro votos más comunes son:
  - **Pluralidad** → selecciona la etiqueta de clase con más votos (**moda** o valor más común).
  - **Mayoritario** → requiere la mitad de los votos (una **mayoría**).
  - **Unánime** → el método requiere que todos los modelos predigan **el mismo valor**; de lo contrario, no se realiza ninguna predicción.

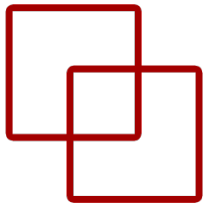
Combinación para clasificación
Votación por etiqueta de clase



- La votación implica que cada modelo que hace una predicción **asigna un voto** a esa clase.
- Un resultado se escoge utilizando los votos o el conteo.
- Hay muchos tipos de votación, los cuatro votos más comunes son:
  - **Pluralidad** → selecciona la etiqueta de clase con más votos (**moda** o valor más común).
  - **Mayoritario** → requiere la mitad de los votos (una **mayoría**).
  - **Unánime** → el método requiere que todos los modelos predigan **el mismo valor**; de lo contrario, no se realiza ninguna predicción.
  - **Ponderado** → **pesa** de alguna manera la predicción realizada por cada modelo.
    - e.g., El peso de cada clasificador se puede establecer proporcionalmente a su rendimiento en un conjunto de validación.

Combinación para clasificación

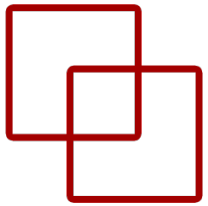
Votación por probabilidades de clase



- Probabilidad de que ocurra un evento como un valor numérico entre 0,0 y 1,0.
  - La suma total de las etiquetas será 1,0.

Combinación para clasificación

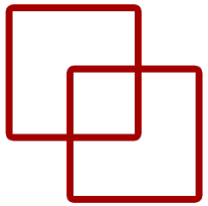
Votación por probabilidades de clase



- Probabilidad de que ocurra un evento como un valor numérico entre 0,0 y 1,0.
  - La suma total de las etiquetas será 1,0.
- Las probabilidades previstas representan el **voto** realizado por cada modelo para cada clase.

Combinación para clasificación

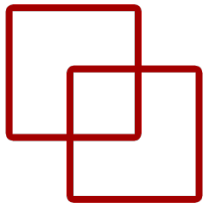
Votación por probabilidades de clase



- Probabilidad de que ocurra un evento como un valor numérico entre 0,0 y 1,0.
  - La suma total de las etiquetas será 1,0.
- Las probabilidades previstas representan el **voto** realizado por cada modelo para cada clase.
- Luego se suman los votos y se puede utilizar un método de votación.

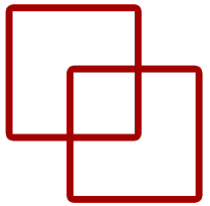
Combinación para clasificación

Votación por probabilidades de clase



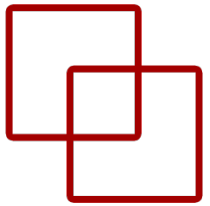
- Probabilidad de que ocurra un evento como un valor numérico entre 0,0 y 1,0.
  - La suma total de las etiquetas será 1,0.
- Las probabilidades previstas representan el **voto** realizado por cada modelo para cada clase.
- Luego se suman los votos y se puede utilizar un método de votación.
- Se tienen los siguientes votos principales:
  - Votar utilizando **probabilidades medias**
  - Votar utilizando **probabilidades de suma**
  - Votar usando **probabilidades de suma ponderada**

Combinación para regresión



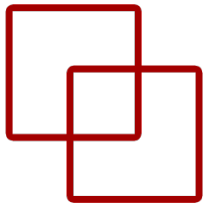
- Implica el uso de métodos estadísticos simples; e.g.:

Combinación para regresión

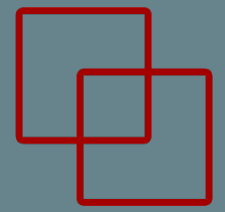


- Implica el uso de métodos estadísticos simples; e.g.:
  - Valor **medio** predicho:
    - Más apropiado cuando la distribución de las predicciones es gaussiana o casi gaussiana.

Combinación para regresión



- Implica el uso de métodos estadísticos simples; e.g.:
  - Valor **medio** predicho:
    - Más apropiado cuando la distribución de las predicciones es gaussiana o casi gaussiana.
  - Valor **mediana** predicho:
    - Más apropiado cuando se desconoce la distribución de las predicciones o no sigue una distribución de probabilidad gaussiana

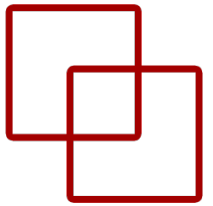


UNED

ETS de Ingeniería Informática

Taxonomía en algoritmos de conjunto

Taxonomías principales

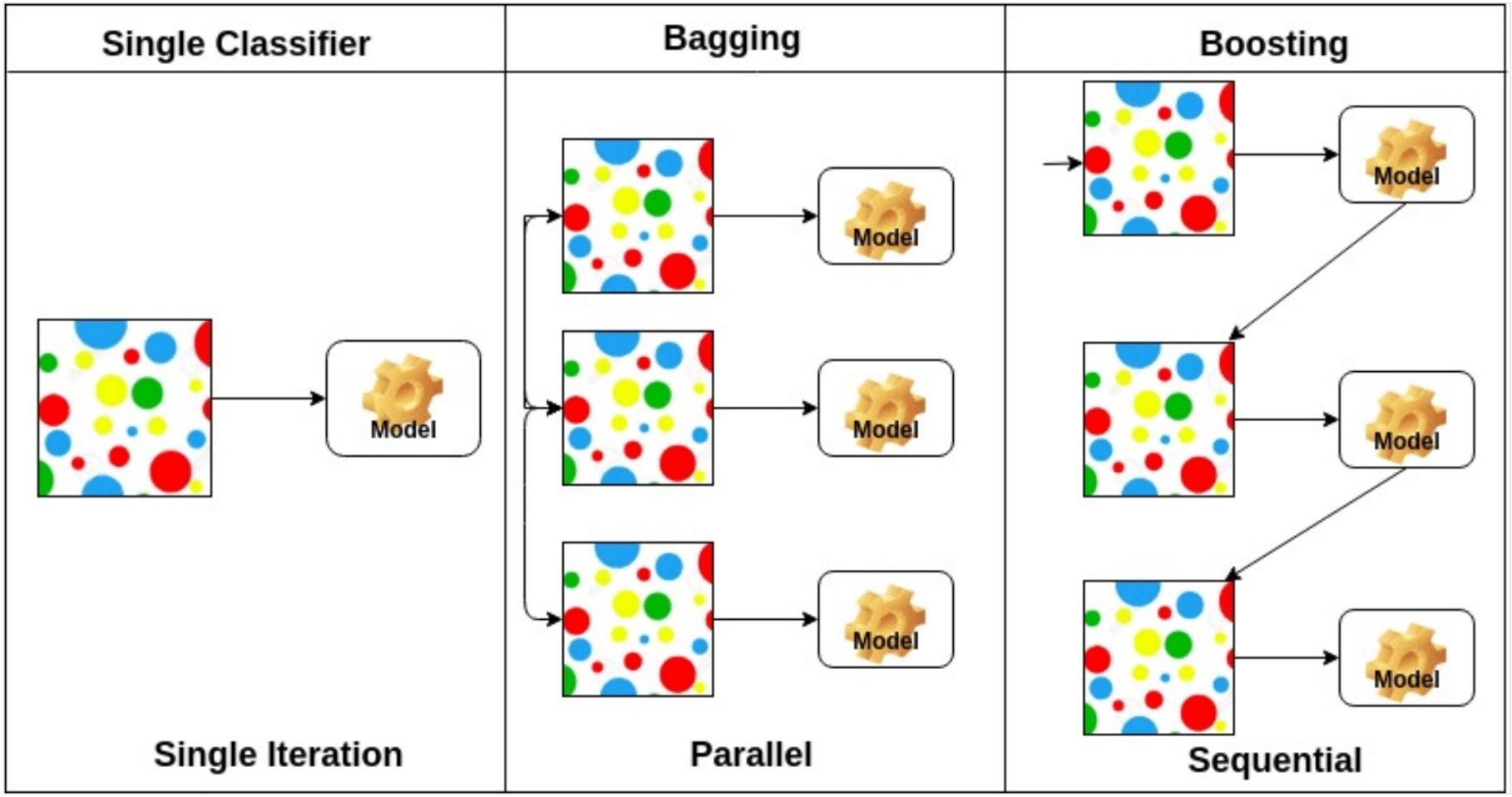


- **Bagging** implica ajustar muchos CART en **diferentes muestras** del mismo conjunto de datos y promediar las predicciones.

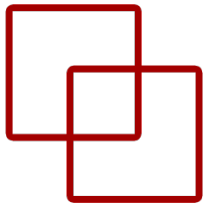
- **Boosting** implica agregar miembros del conjunto **secuencialmente** que **corrigen los errores** de predicción realizado por modelos anteriores y genera un promedio ponderado de las predicciones.

- **Stacking** implica ajustar muchos tipos de **modelos diferentes a los mismos datos** y usar **otro modelo** para aprender cómo combinar mejor las predicciones.

Algoritmos de conjunto

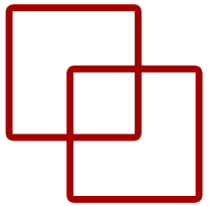


Bagging



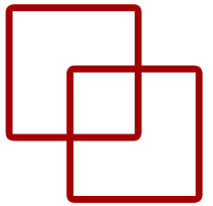
- Construye múltiples modelos (típicamente modelos del mismo tipo) a partir de diferentes submuestras del conjunto de datos de entrenamiento.

Bagging



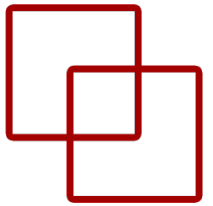
- Construye múltiples modelos (típicamente modelos del mismo tipo) a partir de diferentes submuestras del conjunto de datos de entrenamiento.
- Las muestras (filas) se extraen del conjunto de datos al azar, aunque con reemplazo (*boostrap sample*).
  - Reemplazo significa que si se selecciona una fila, se devuelve al conjunto de datos de entrenamiento para una posible nueva selección en la misma muestra de entrenamiento (se puede seleccionar cero, una o varias veces).

Bagging



- Construye múltiples modelos (típicamente modelos del mismo tipo) a partir de diferentes submuestras del conjunto de datos de entrenamiento.
- Las muestras (filas) se extraen del conjunto de datos al azar, aunque con reemplazo (*boostrap sample*).
  - Reemplazo significa que si se selecciona una fila, se devuelve al conjunto de datos de entrenamiento para una posible nueva selección en la misma muestra de entrenamiento (se puede seleccionar cero, una o varias veces).
- Elementos clave en Bagging
  - Muestras boostrap del conjunto de datos de entrenamiento.
  - Los árboles de decisión no podados se ajustan a cada muestra.
  - Votación simple o promediación de predicciones.

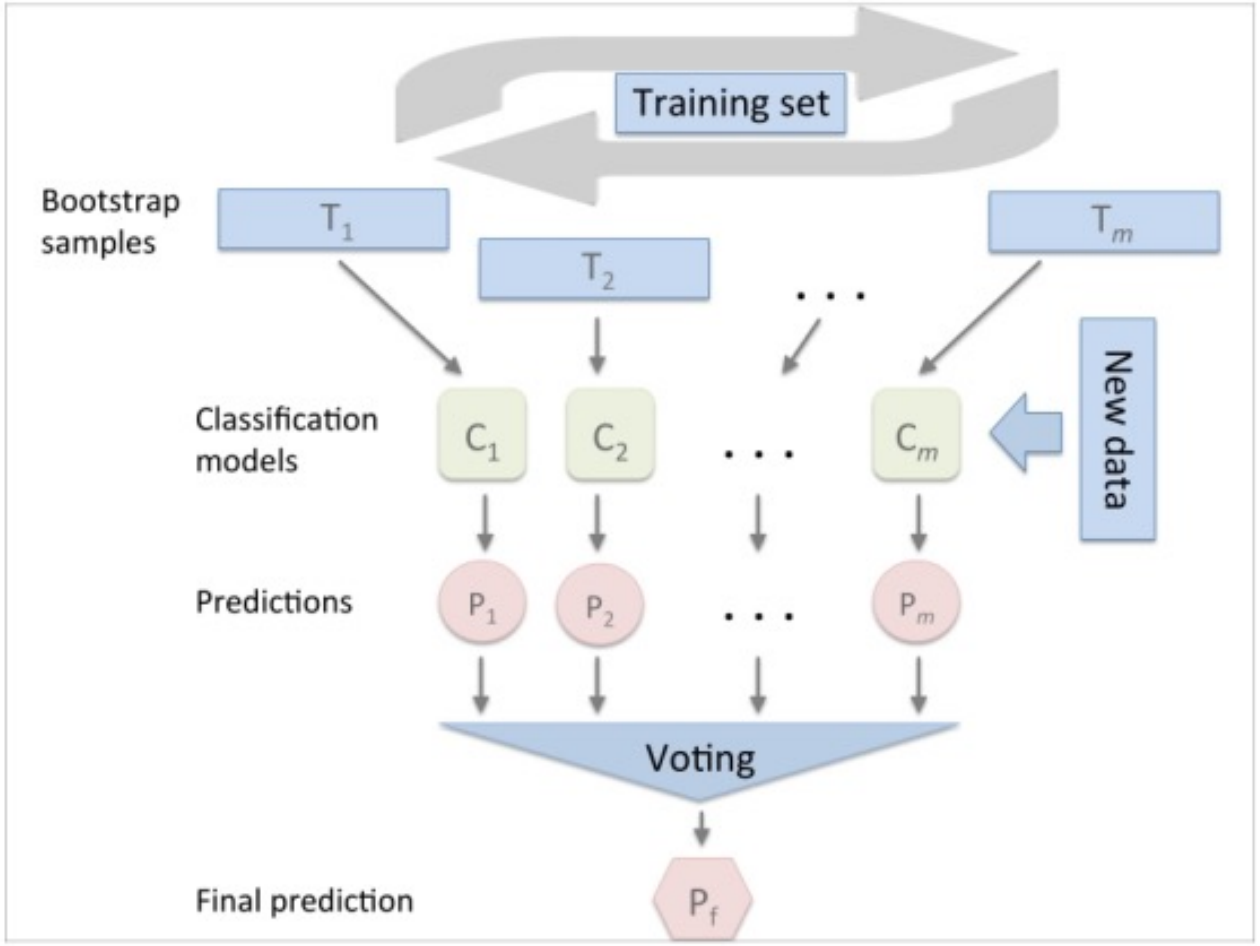
Bagging



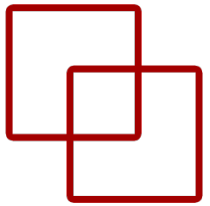
- Construye múltiples modelos (típicamente modelos del mismo tipo) a partir de diferentes submuestras del conjunto de datos de entrenamiento.
- Las muestras (filas) se extraen del conjunto de datos al azar, aunque con reemplazo (*boostrap sample*).
  - Reemplazo significa que si se selecciona una fila, se devuelve al conjunto de datos de entrenamiento para una posible nueva selección en la misma muestra de entrenamiento (se puede seleccionar cero, una o varias veces).
- Elementos clave en Bagging
  - Muestras boostrap del conjunto de datos de entrenamiento.
  - Los árboles de decisión no podados se ajustan a cada muestra.
  - Votación simple o promediación de predicciones.
- La contribución está en la **variación de los datos de entrenamiento** utilizados para ajustar cada miembro del conjunto, dando modelos hábiles pero diferentes.

Bagging

- Se pueden introducir más cambios en el dataset *train*
- Se puede reemplazar el ajuste del algoritmo en *train* y
- Se puede modificar el mecanismo utilizado para combinar predicciones
- Algoritmos:
  - Bagged Decision Trees
  - Random Forest
  - Extra Trees

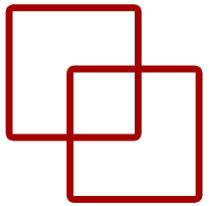


Boosting



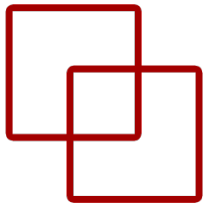
- Construye múltiples modelos (típicamente modelos del mismo tipo) secuenciales, cada uno de los cuales aprende a **corregir los errores de predicción** de un modelo anterior en la cadena.

Boosting



- Construye múltiples modelos (típicamente modelos del mismo tipo) secuenciales, cada uno de los cuales aprende a **corregir los errores de predicción** de un modelo anterior en la cadena.
- El dataset ***train*** **no se modifica**.

Boosting



- Construye múltiples modelos (típicamente modelos del mismo tipo) secuenciales, cada uno de los cuales aprende a **corregir los errores de predicción** de un modelo anterior en la cadena.
- El dataset ***train*** **no se modifica**.
- El objetivo es desarrollar **un modelo fuerte a partir de muchos modelos débiles** especialmente diseñados que se combinan mediante votación simple o promediando.

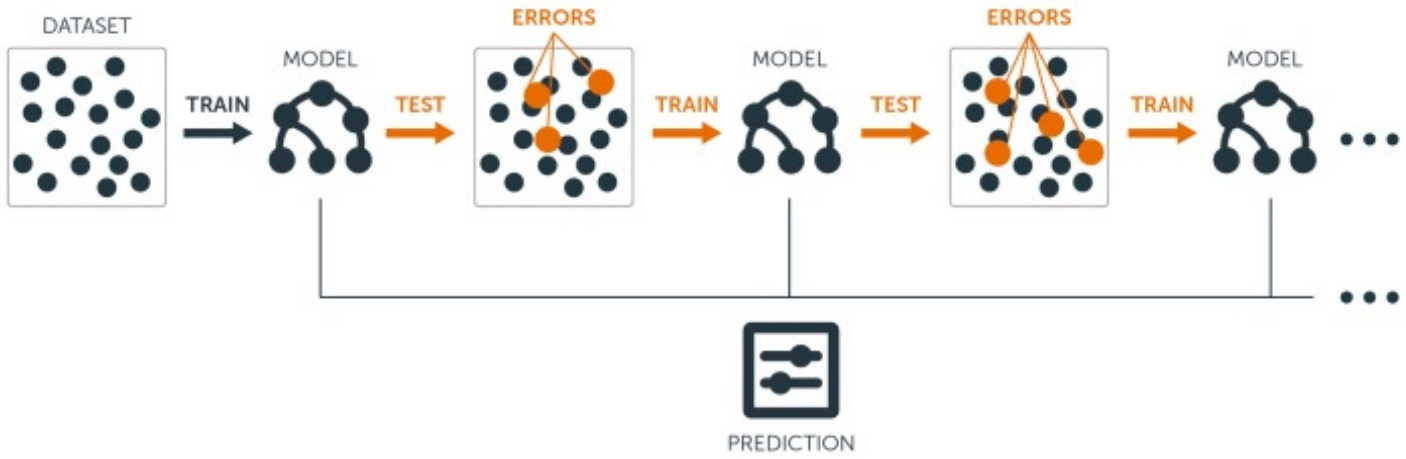
Boosting

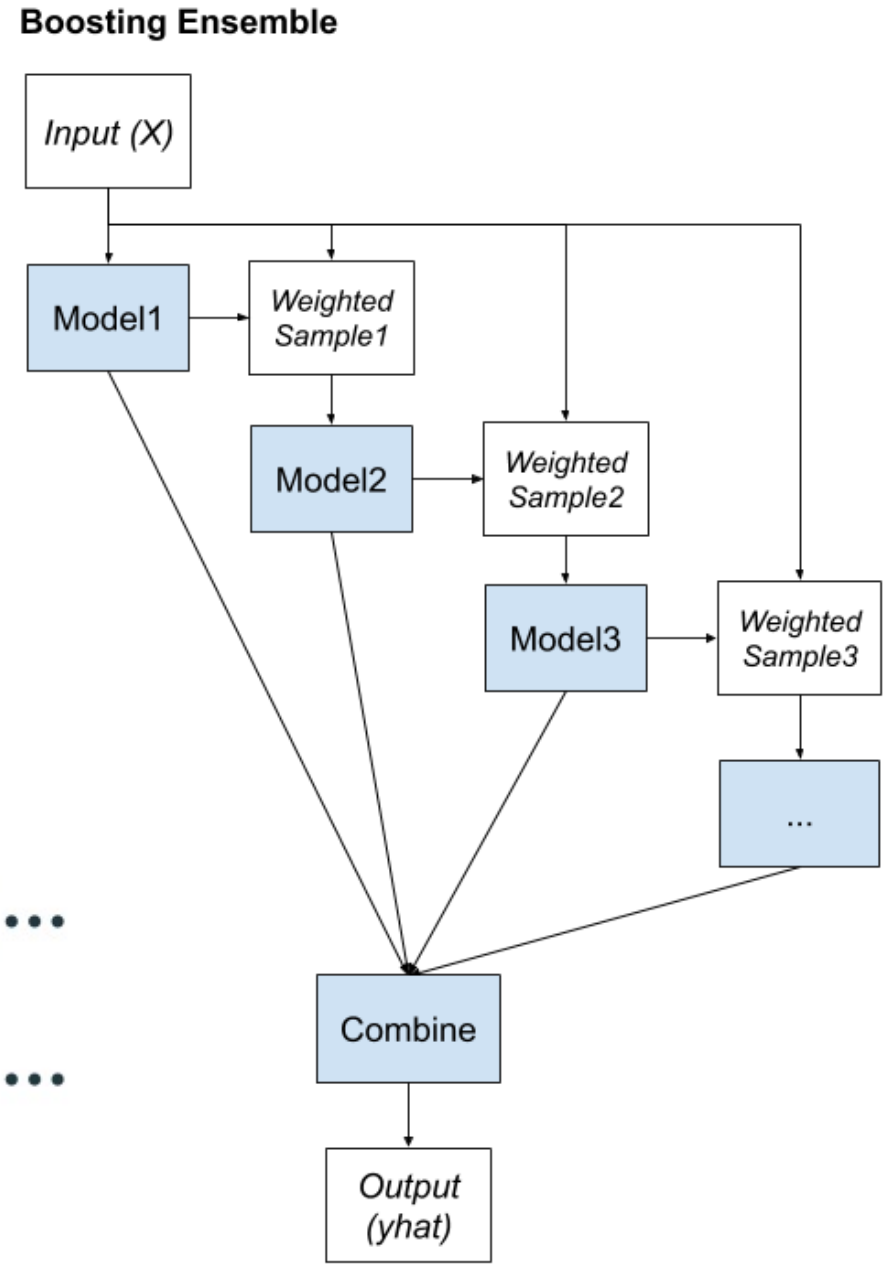
- Construye múltiples modelos (típicamente modelos del mismo tipo) secuenciales, cada uno de los cuales aprende a **corregir los errores de predicción** de un modelo anterior en la cadena.
- El dataset ***train*** **no se modifica**.
- El objetivo es desarrollar **un modelo fuerte a partir de muchos modelos débiles** especialmente diseñados que se combinan mediante votación simple o promediando.
- Elementos clave de Boosting:
  - **Sesgar** los datos de entrenamiento hacia aquellos ejemplos que son difíciles de predecir.
  - **Agregar** iterativamente miembros del conjunto para corregir predicciones de modelos anteriores.
  - **Combinar** predicciones utilizando un promedio ponderado de modelos.

Boosting

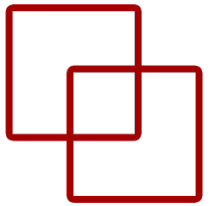
- Construye múltiples modelos (típicamente modelos del mismo tipo) secuenciales, cada uno de los cuales aprende a **corregir los errores de predicción** de un modelo anterior en la cadena.
- El dataset ***train*** **no se modifica**.
- El objetivo es desarrollar **un modelo fuerte a partir de muchos modelos débiles** especialmente diseñados que se combinan mediante votación simple o promediando.
- Elementos clave de Boosting:
  - **Sesgar** los datos de entrenamiento hacia aquellos ejemplos que son difíciles de predecir.
  - **Agregar** iterativamente miembros del conjunto para corregir predicciones de modelos anteriores.
  - **Combinar** predicciones utilizando un promedio ponderado de modelos.
- Algoritmos: AdaBoost, Gradient Boosting Machines, XGBoost

Boosting


DATASET
TRAIN
MODEL
TEST
ERRORS
TRAIN
MODEL
TEST
ERRORS
TRAIN
MODEL
PREDICTION

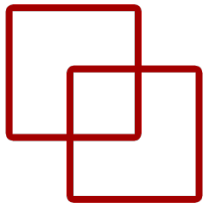

Boosting Ensemble
Input (X)
Model1
Weighted Sample1
Model2
Weighted Sample2
Model3
Weighted Sample3
...
Combine
Output (yhat)

Stacking



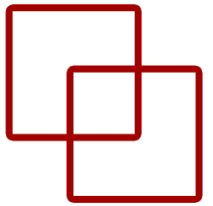
- Construye múltiples modelos, típicamente modelos de diferentes tipos (modelos de **nivel 0**); y un modelo supervisado (modelo de **nivel 1**) que aprende cómo combinar mejor las predicciones de los modelos primarios.

Stacking



- Construye múltiples modelos, típicamente modelos de diferentes tipos (modelos de **nivel 0**); y un modelo supervisado (modelo de **nivel 1**) que aprende cómo combinar mejor las predicciones de los modelos primarios.
- Se pueden usar **más niveles** e integración entre ellos, i.e., puede haber más de un metaclasificador.

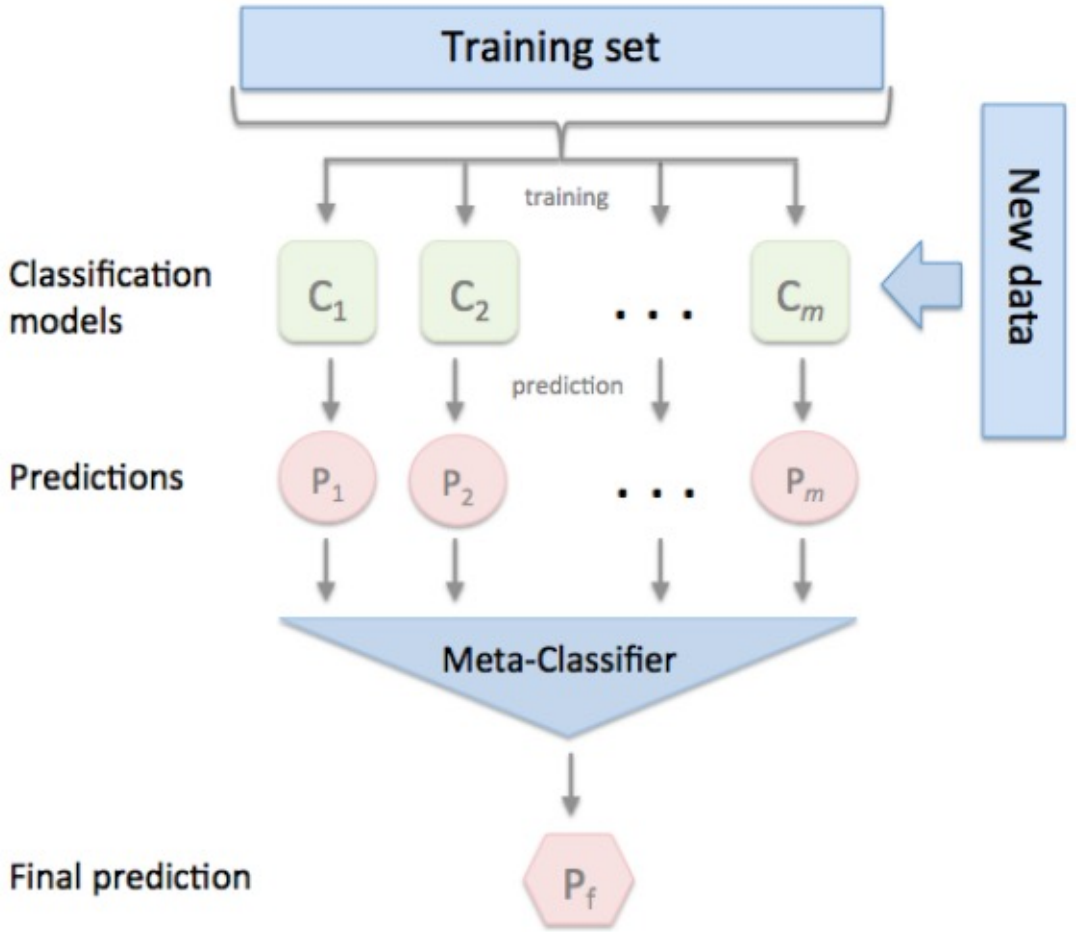
Stacking

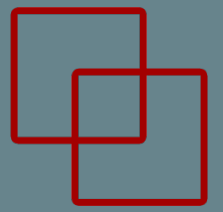


- Construye múltiples modelos, típicamente modelos de diferentes tipos (modelos de **nivel 0**); y un modelo supervisado (modelo de **nivel 1**) que aprende cómo combinar mejor las predicciones de los modelos primarios.
- Se pueden usar **más niveles** e integración entre ellos, i.e., puede haber más de un metaclasificador.
- Elementos clave de Stacking:
  - Conjunto de datos de entrenamiento sin cambios.
  - Diferentes algoritmos de aprendizaje automático para cada miembro del conjunto.
  - Modelo de aprendizaje automático para aprender cómo combinar mejor las predicciones.

Stacking

- El metaclasicador más común es que sea una regresión lineal o logística.
- Algoritmos:
  - Stacked Generalization
  - Blending Ensemble
  - Super Learner Ensemble



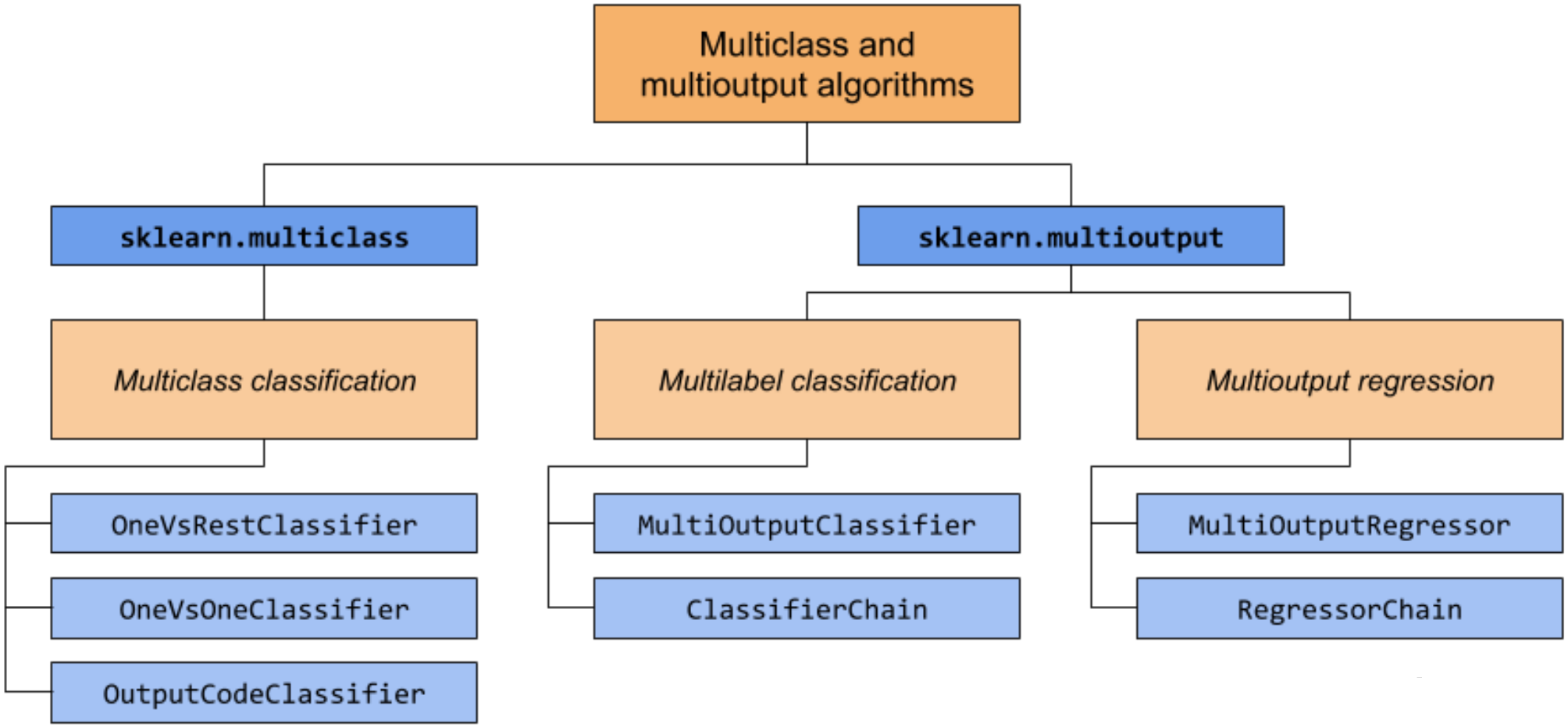


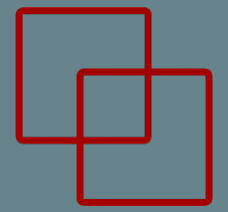
UNED

ETS de Ingeniería Informática

Modelos Múltiples

Background


Multiclass and multioutput algorithms
sklearn.multiclass
sklearn.multioutput
Multiclass classification
Multilabel classification
Multioutput regression
OneVsRestClassifier
OneVsOneClassifier
OutputCodeClassifier
MultiOutputClassifier
ClassifierChain
MultiOutputRegressor
RegressorChain

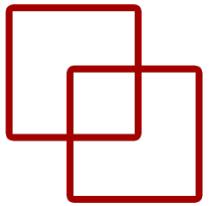


UNED

ETS de Ingeniería Informática

One-Vs-One & One-Vs-Rest

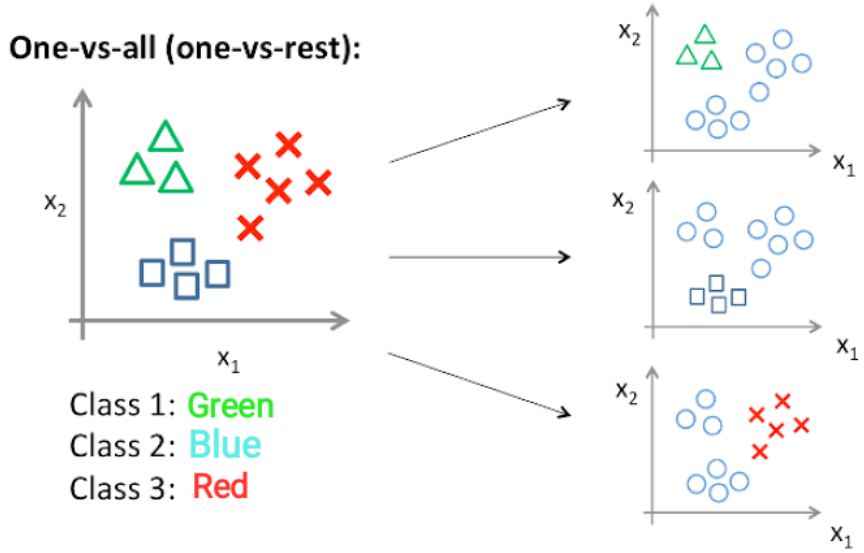
One-Vs-One & One-Vs-Rest



- Los modelos de clasificación binaria como la LoR y la SVM **no admiten la clasificación multiclase** de forma nativa y requieren metaestrategias.

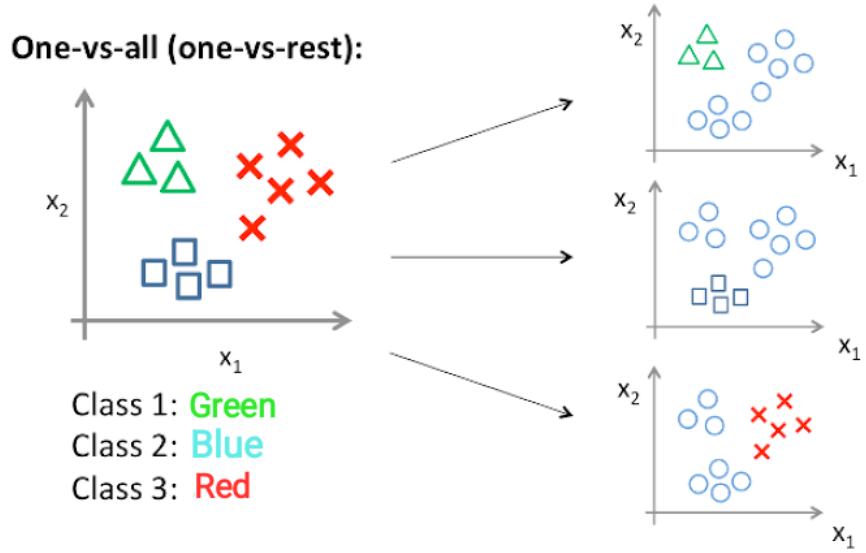
One-Vs-One & One-Vs-Rest

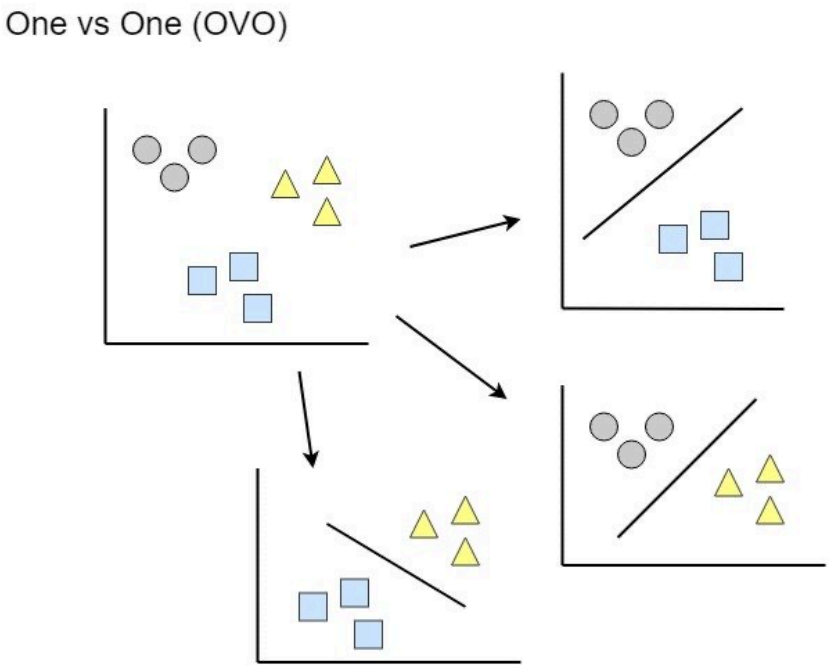
- Los modelos de clasificación binaria como la LoR y la SVM **no admiten la clasificación multiclase** de forma nativa y requieren metaestrategias.
- La estrategia **One-vs-Rest (OVR)** divide una clasificación multiclase en un problema de **clasificación binaria por clase**.

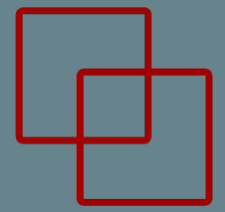


One-Vs-One & One-Vs-Rest

- Los modelos de clasificación binaria como la LoR y la SVM **no admiten la clasificación multiclase** de forma nativa y requieren metaestrategias.
- La estrategia **One-vs-Rest (OVR)** divide una clasificación multiclase en un problema de **clasificación binaria por clase**.
- La estrategia **One-vs-One (OVO)** divide una clasificación multiclase en un problema de **clasificación binaria por cada par de clases**.





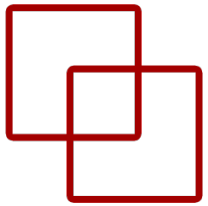


UNED

ETS de Ingeniería Informática

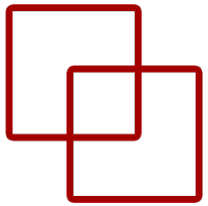
Códigos de salida con corrección de errores

Códigos de salida con corrección de errores



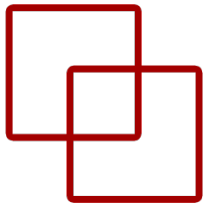
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.

Códigos de salida con corrección de errores



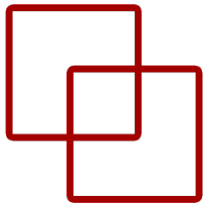
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.
- Cada **bit** de la cadena se puede predecir mediante un problema de clasificación binaria.
    - Arbitrariamente, se pueden elegir codificaciones de longitud para un problema de clasificación multiclase.

Códigos de salida con corrección de errores



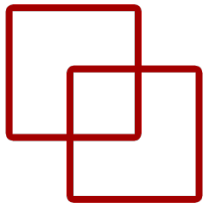
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.
- Cada **bit** de la cadena se puede predecir mediante un problema de clasificación binaria.
    - Arbitrariamente, se pueden elegir codificaciones de longitud para un problema de clasificación multiclase.
- Cada modelo recibe el patrón de entrada completo y predice una posición en la cadena de salida.

Códigos de salida con corrección de errores



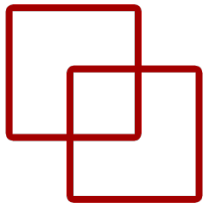
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.
- Cada **bit** de la cadena se puede predecir mediante un problema de clasificación binaria.
    - Arbitrariamente, se pueden elegir codificaciones de longitud para un problema de clasificación multiclase.
- Cada modelo recibe el patrón de entrada completo y predice una posición en la cadena de salida.

Códigos de salida con corrección de errores



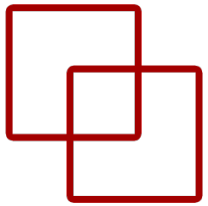
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.
- Cada **bit** de la cadena se puede predecir mediante un problema de clasificación binaria.
    - Arbitrariamente, se pueden elegir codificaciones de longitud para un problema de clasificación multiclase.
- Cada modelo recibe el patrón de entrada completo y predice una posición en la cadena de salida.
- Durante el entrenamiento, cada modelo se puede entrenar para producir la salida 0 o 1 para la tarea de clasificación binaria.

Códigos de salida con corrección de errores



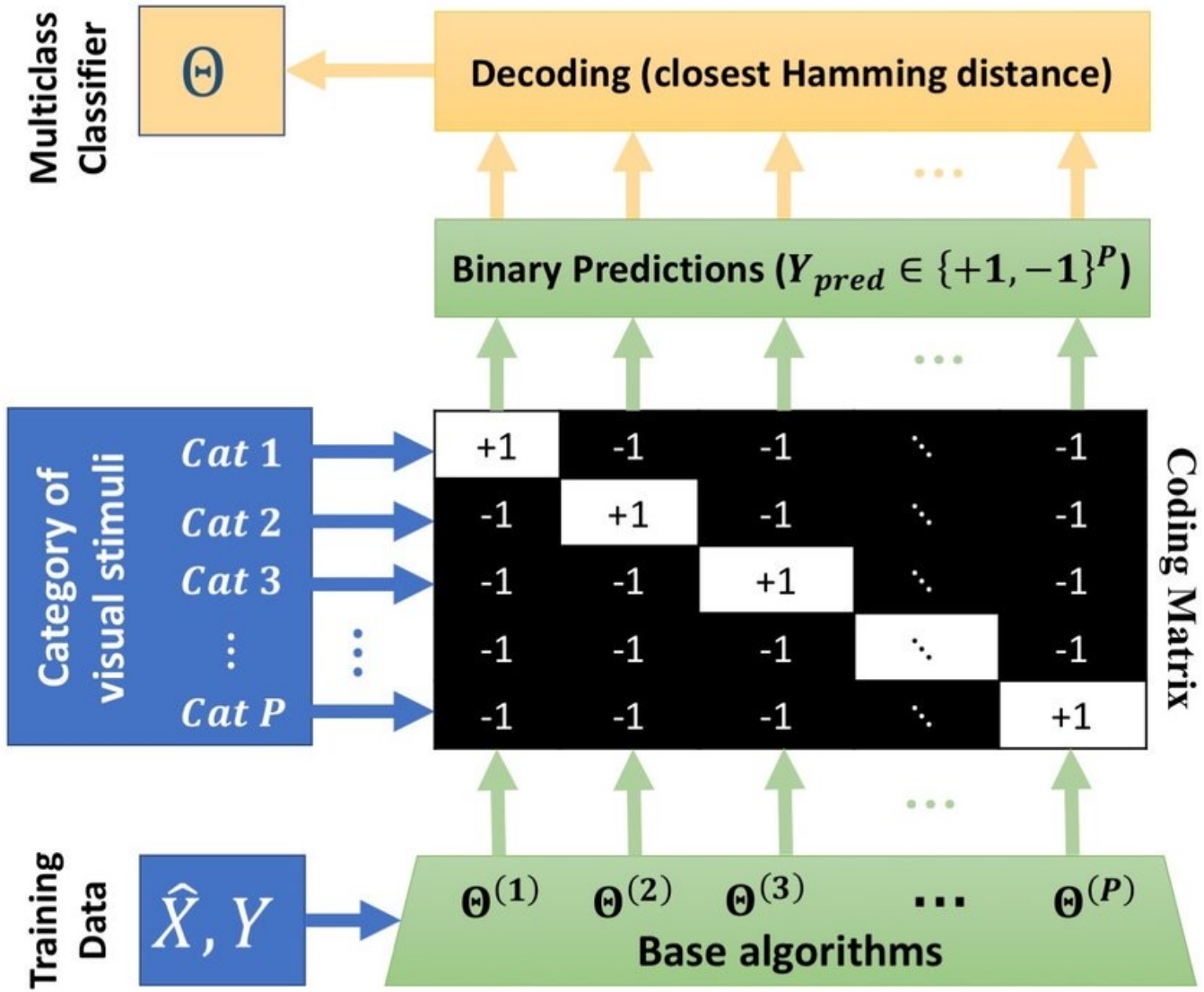
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.
- Cada **bit** de la cadena se puede predecir mediante un problema de clasificación binaria.
    - Arbitrariamente, se pueden elegir codificaciones de longitud para un problema de clasificación multiclase.
- Cada modelo recibe el patrón de entrada completo y predice una posición en la cadena de salida.
- Durante el entrenamiento, cada modelo se puede entrenar para producir la salida 0 o 1 para la tarea de clasificación binaria.
- Luego se puede hacer una predicción para nuevos ejemplos utilizando cada modelo para hacer una predicción de la entrada para crear la cadena binaria y luego **comparar** la cadena binaria con la codificación conocida de cada clase.

Códigos de salida con corrección de errores



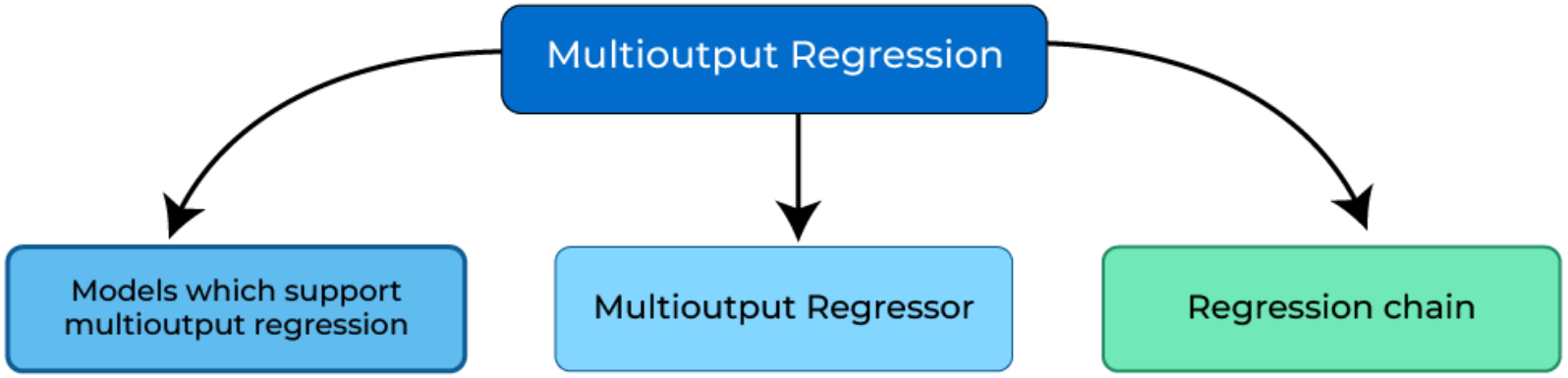
- Un enfoque relacionado a OvR y OvO consiste en preparar una **codificación binaria** (e.g., una cadena de bits) para representar cada clase del problema.
- Cada **bit** de la cadena se puede predecir mediante un problema de clasificación binaria.
    - Arbitrariamente, se pueden elegir codificaciones de longitud para un problema de clasificación multiclase.
- Cada modelo recibe el patrón de entrada completo y predice una posición en la cadena de salida.
- Durante el entrenamiento, cada modelo se puede entrenar para producir la salida 0 o 1 para la tarea de clasificación binaria.
- Luego se puede hacer una predicción para nuevos ejemplos utilizando cada modelo para hacer una predicción de la entrada para crear la cadena binaria y luego **comparar** la cadena binaria con la codificación conocida de cada clase.
- Luego se elige como salida la codificación de clase que tiene la **menor distancia** a la predicción.

Códigos de salida con corrección de errores

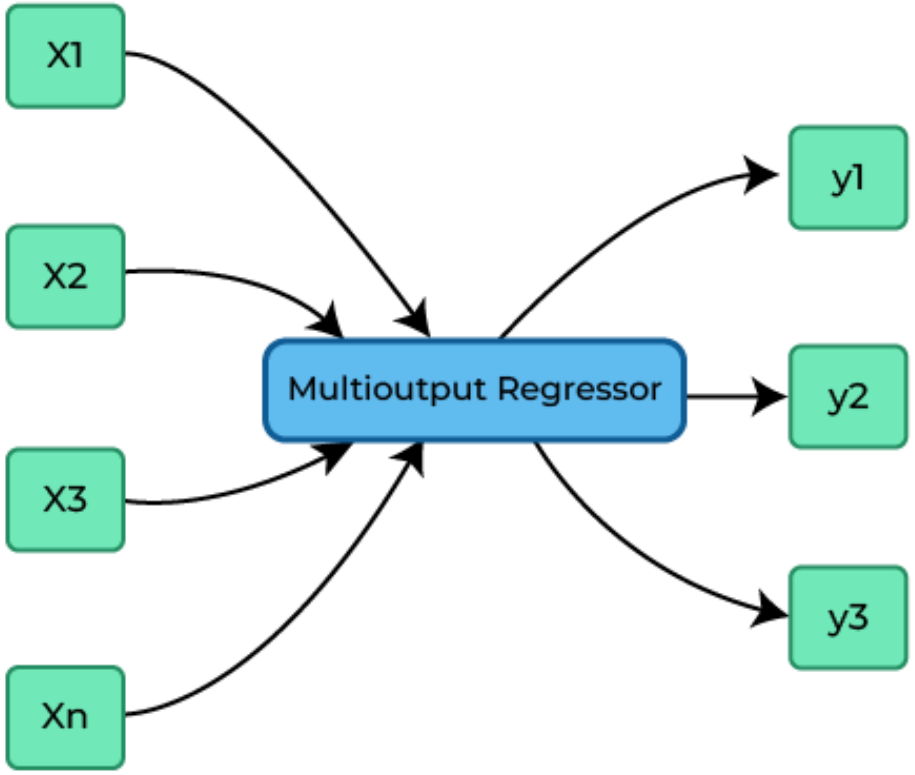


Modelos de regresión salidas múlples

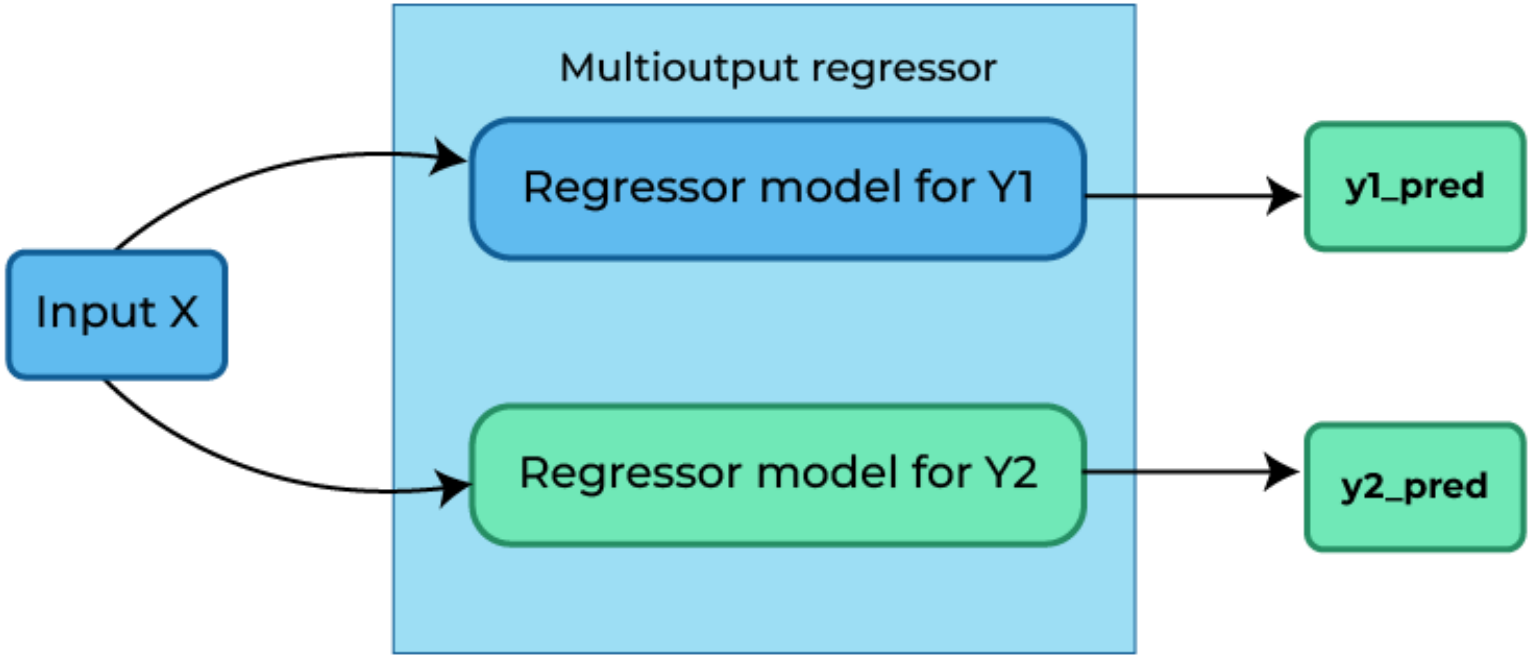
Modelos de regresión salidas múlples



Regresión multisalida dependiente



Regresión multisalida independiente



Mezcla de expertos (MoE)

Subtareas y expertos

- Podemos **dividir** el espacio de características de entrada en subespacios según algún conocimiento de dominio del problema.

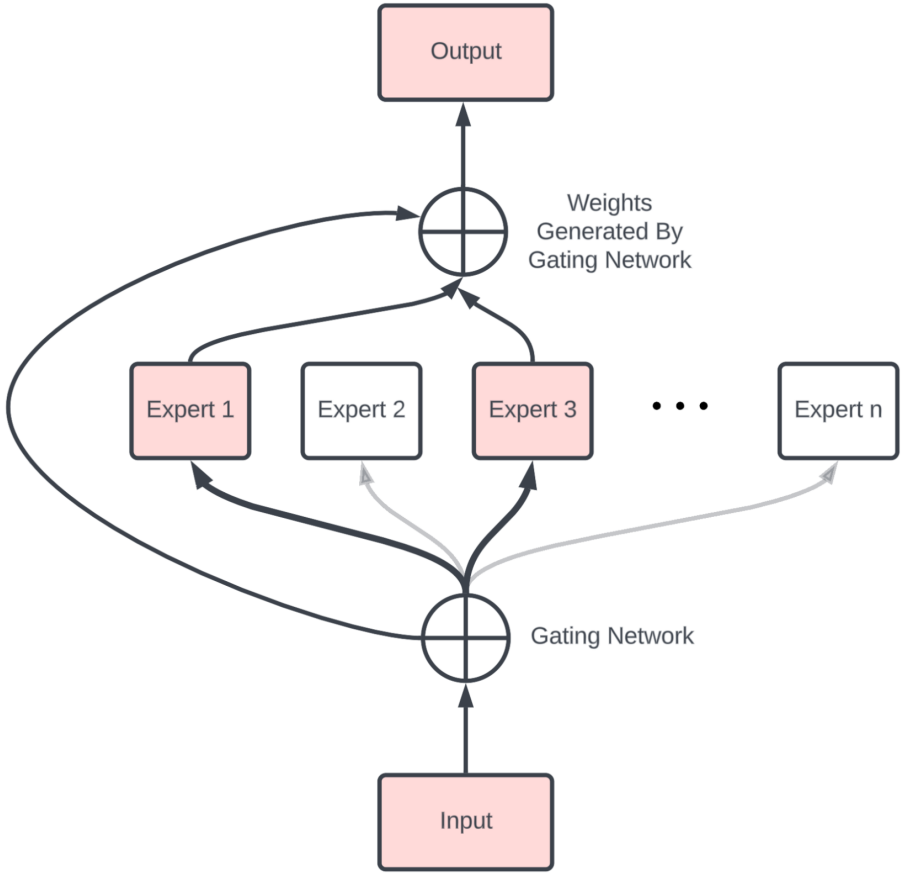


Figura tomada de: https://deepgram.com/learn/mixture-of-experts-ml-model-guide

Subtareas y expertos

- Podemos **dividir** el espacio de características de entrada en subespacios según algún conocimiento de dominio del problema.
- Luego se puede **entrenar un modelo en cada subespacio** del problema, convirtiéndose de hecho en un experto en el subproblema específico.

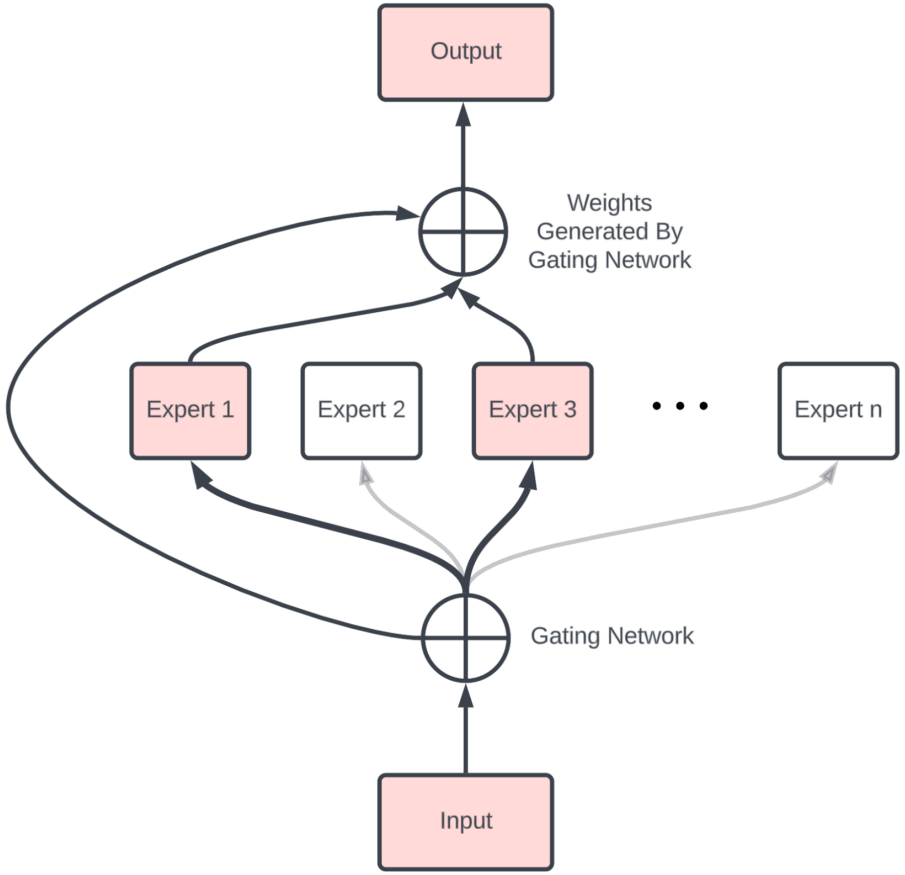


Figura tomada de: https://deepgram.com/learn/mixture-of-experts-ml-model-guide

Subtareas y expertos

- Podemos **dividir** el espacio de características de entrada en subespacios según algún conocimiento de dominio del problema.
- Luego se puede **entrenar un modelo en cada subespacio** del problema, convirtiéndose de hecho en un experto en el subproblema específico.
- Luego, **un modelo aprende a qué experto** recurrir para predecir nuevos ejemplos en el futuro.

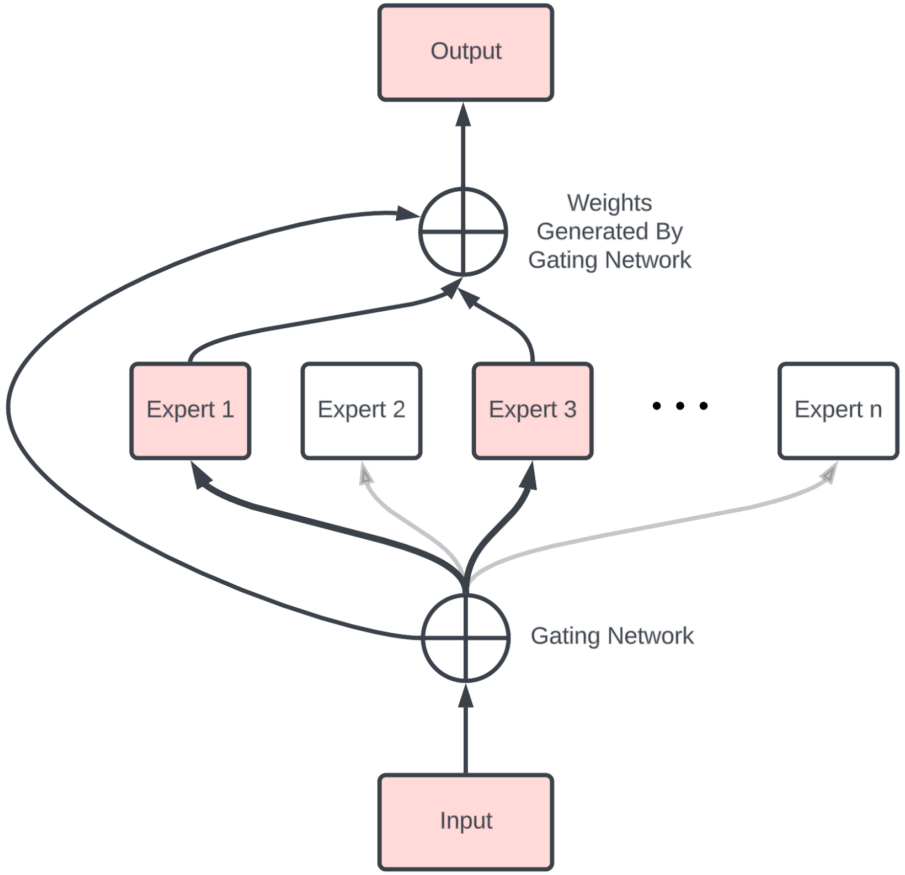


Figura tomada de: https://deepgram.com/learn/mixture-of-experts-ml-model-guide

Subtareas y expertos

- Podemos **dividir** el espacio de características de entrada en subespacios según algún conocimiento de dominio del problema.
- Luego se puede **entrenar un modelo en cada subespacio** del problema, convirtiéndose de hecho en un experto en el subproblema específico.
- Luego, **un modelo aprende a qué experto** recurrir para predecir nuevos ejemplos en el futuro.
- Los **subproblemas pueden superponerse o no**, y los expertos de subproblemas similares o relacionados pueden contribuir a los ejemplos que técnicamente están fuera de su experiencia.

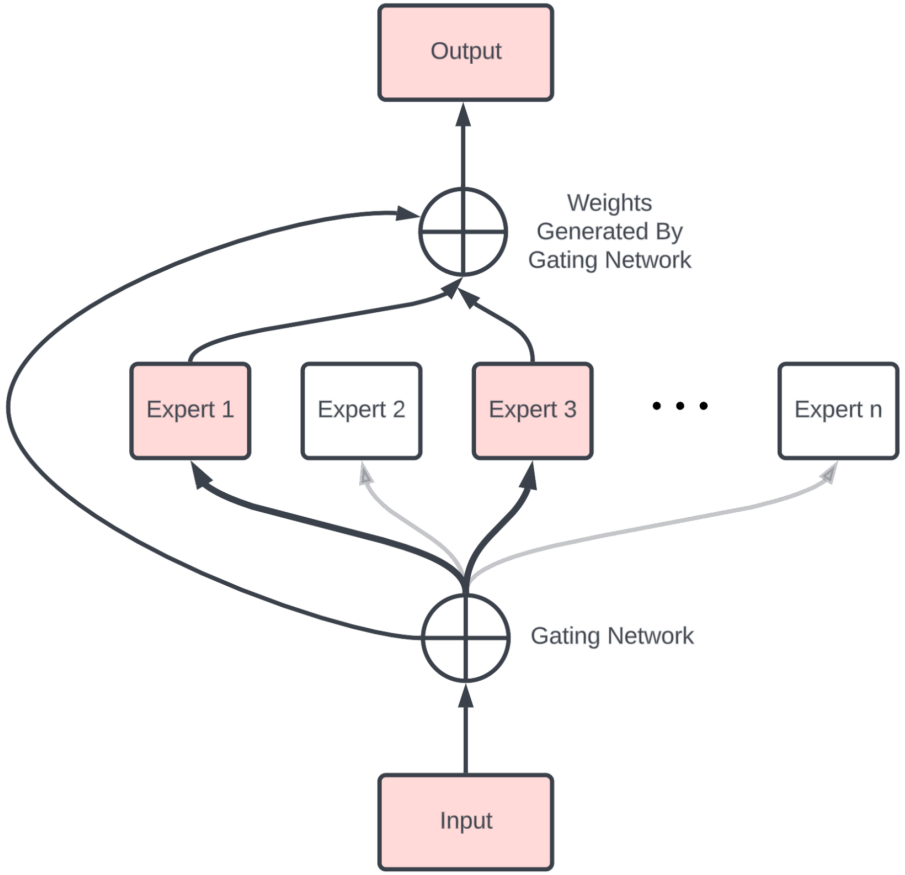


Figura tomada de: https://deepgram.com/learn/mixture-of-experts-ml-model-guide

Mezcla de expertos

- Hay cuatro elementos para el enfoque, que son:
  - **División** de una tarea en subtareas.
  - **Desarrollar** un experto para cada subtarea.
  - Utilizar un **modelo de activación** para decidir qué experto utilizar.
  - **Agrupar predicciones** y generar resultados del modelo de activación para realizar una predicción.

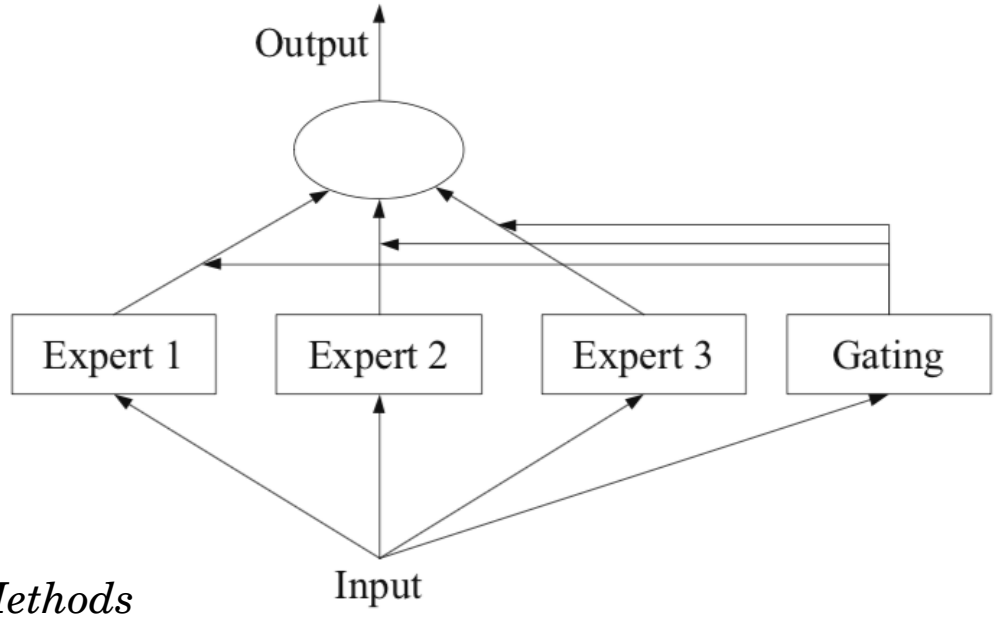
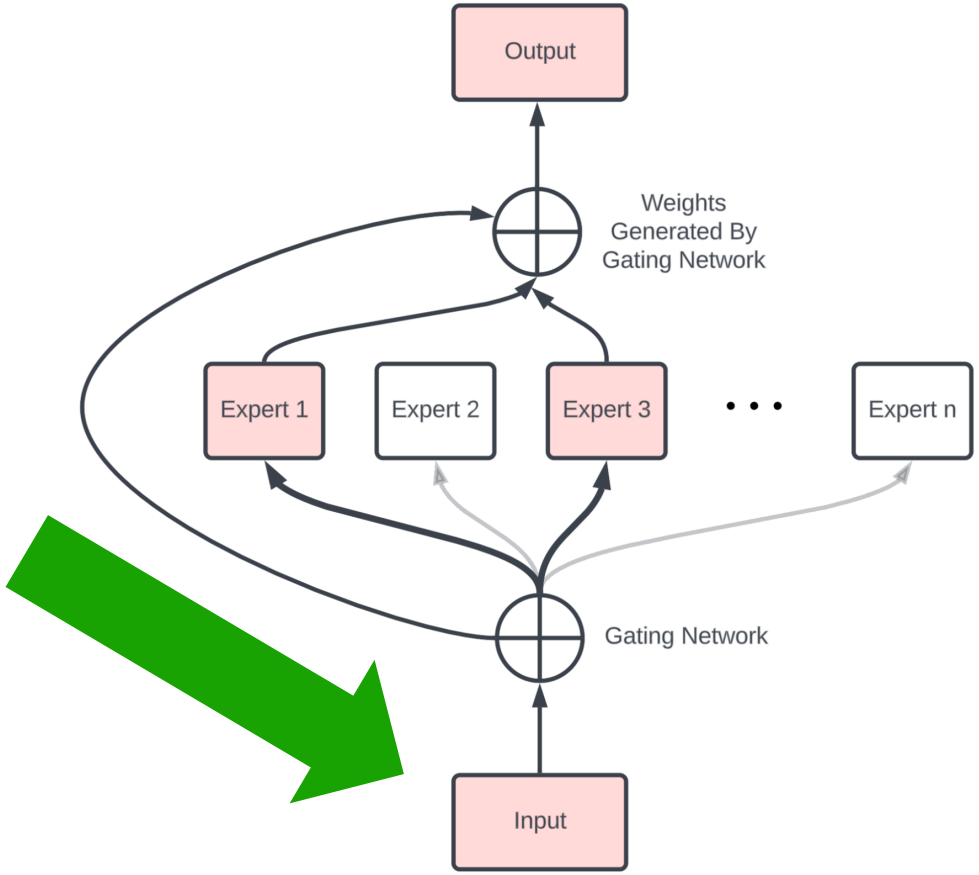


Figura tomada del libro *Ensemble Methods*

Mezcla de expertos

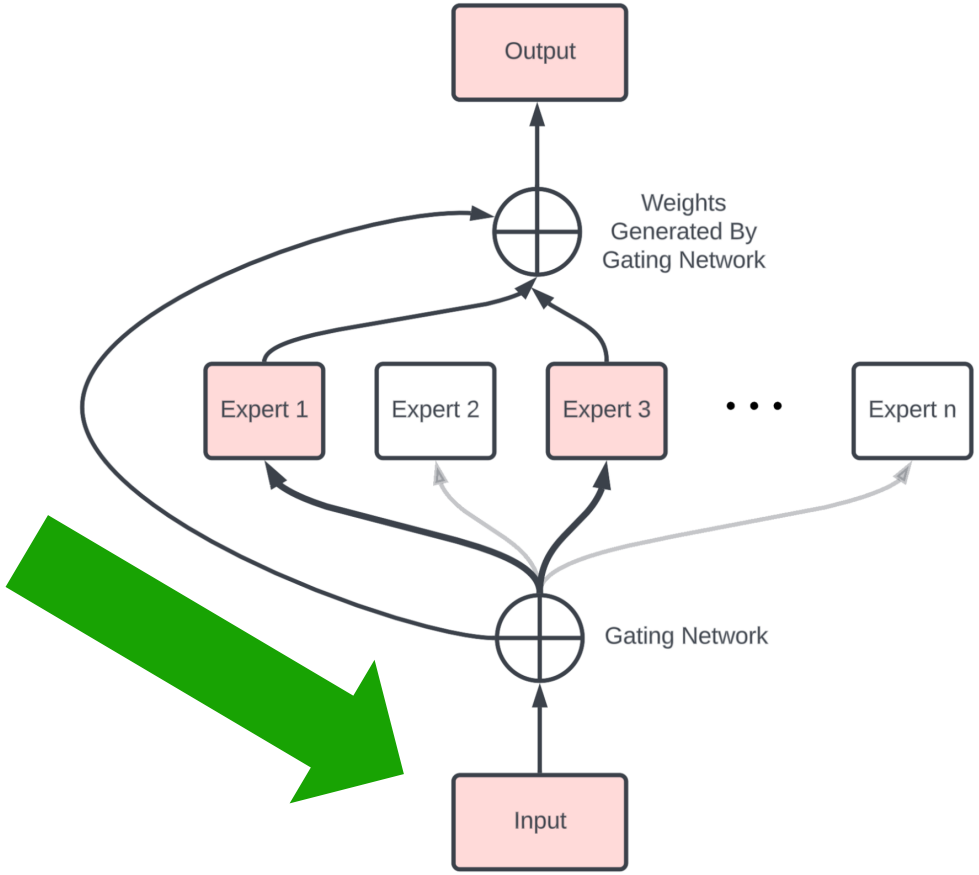
1. Subtareas

- El primer paso es dividir el problema del modelado predictivo en **subtareas**.
  - Por ejemplo, una imagen podría dividirse en elementos separados como fondo, primer plano, objetos, colores, líneas, etc.



Mezcla de expertos
1. Subtareas

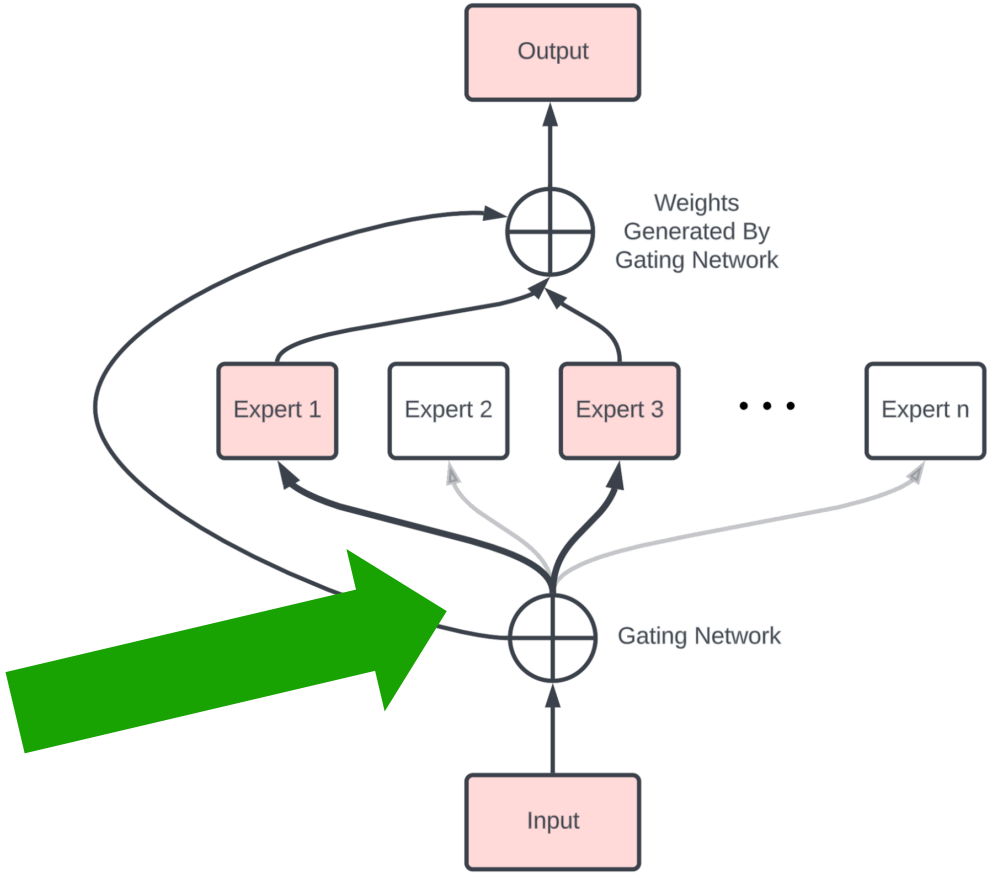
- El primer paso es dividir el problema del modelado predictivo en **subtareas**.
  - Por ejemplo, una imagen podría dividirse en elementos separados como fondo, primer plano, objetos, colores, líneas, etc.
- Para aquellos problemas en los que la división de la tarea en subtareas no es obvia, se podría utilizar un enfoque más simple y genérico.
  - Por ejemplo, se podría imaginar un enfoque que divida el espacio de características de entrada en **grupos de columnas o separe ejemplos en el espacio de características** en función de medidas de distancia, valores internos y atípicos para una distribución estándar, y mucho más.



Mezcla de expertos

2. Modelos expertos

- A continuación, se **diseña un experto** para cada subtarea.

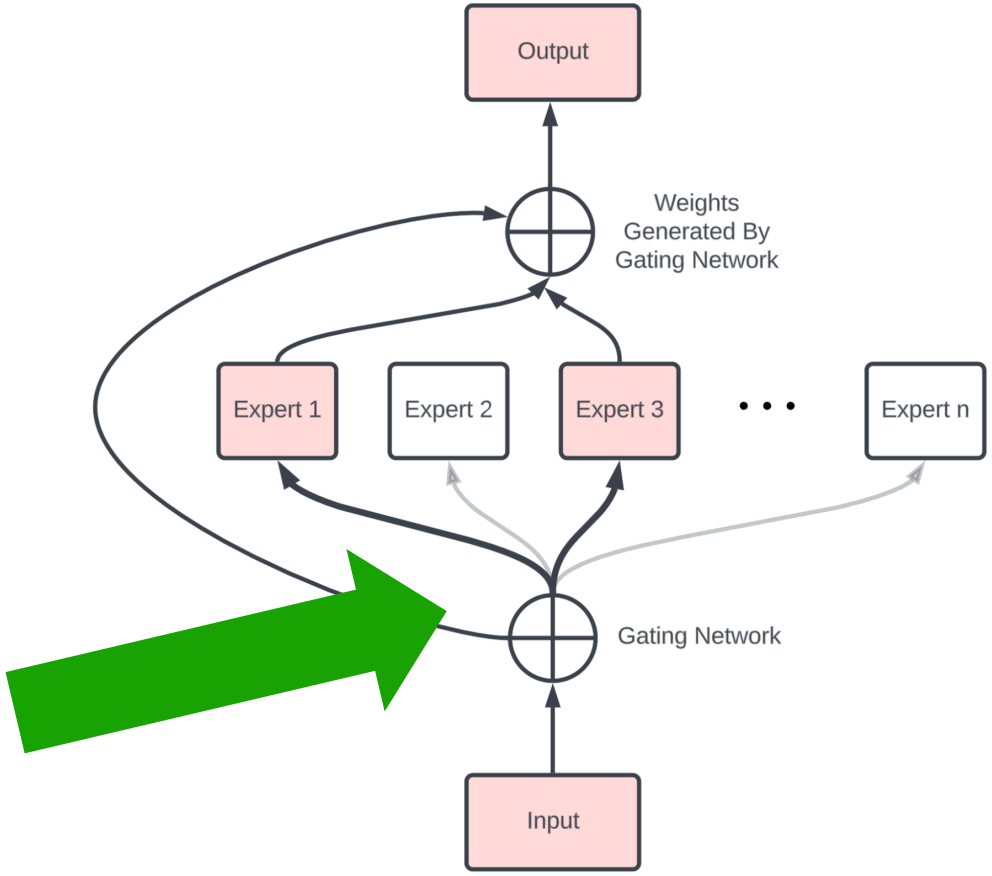


Mezcla de expertos

2. Modelos expertos

- A continuación, se **diseña un experto** para cada subtarea.

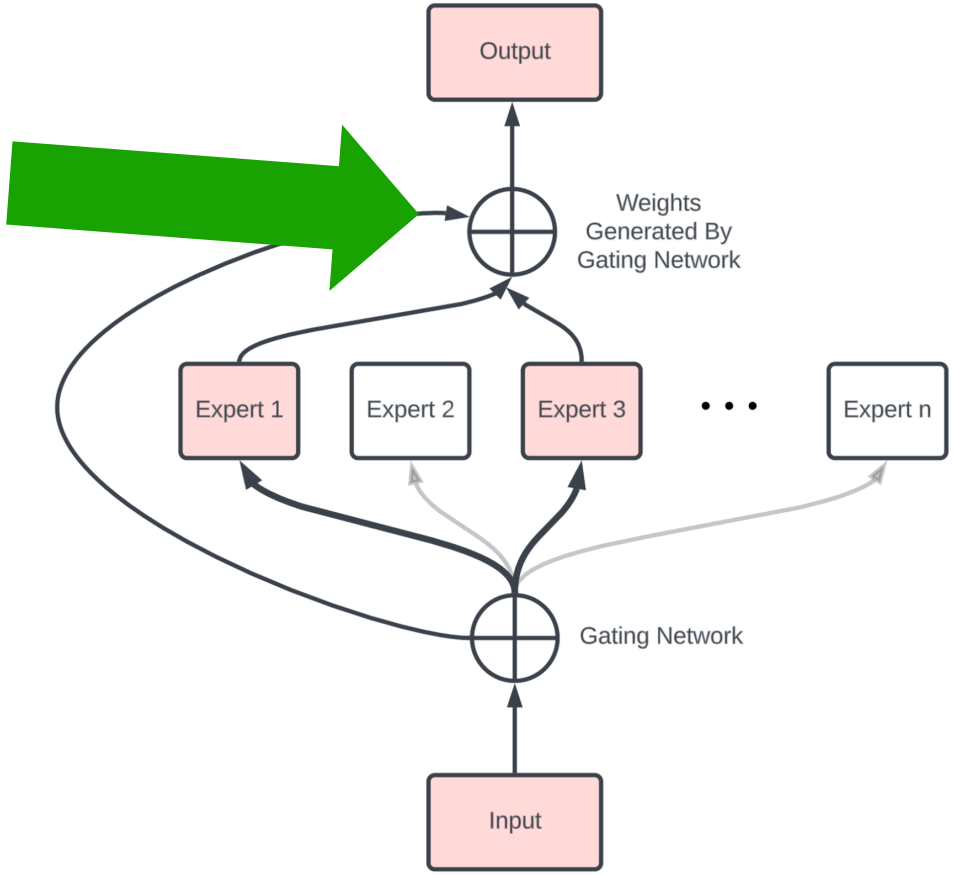
- Cada uno de los expertos recibe el mismo patrón de entrada (fila) y hace una predicción.



Mezcla de expertos

3. Modelo de compuerta (Gating model)

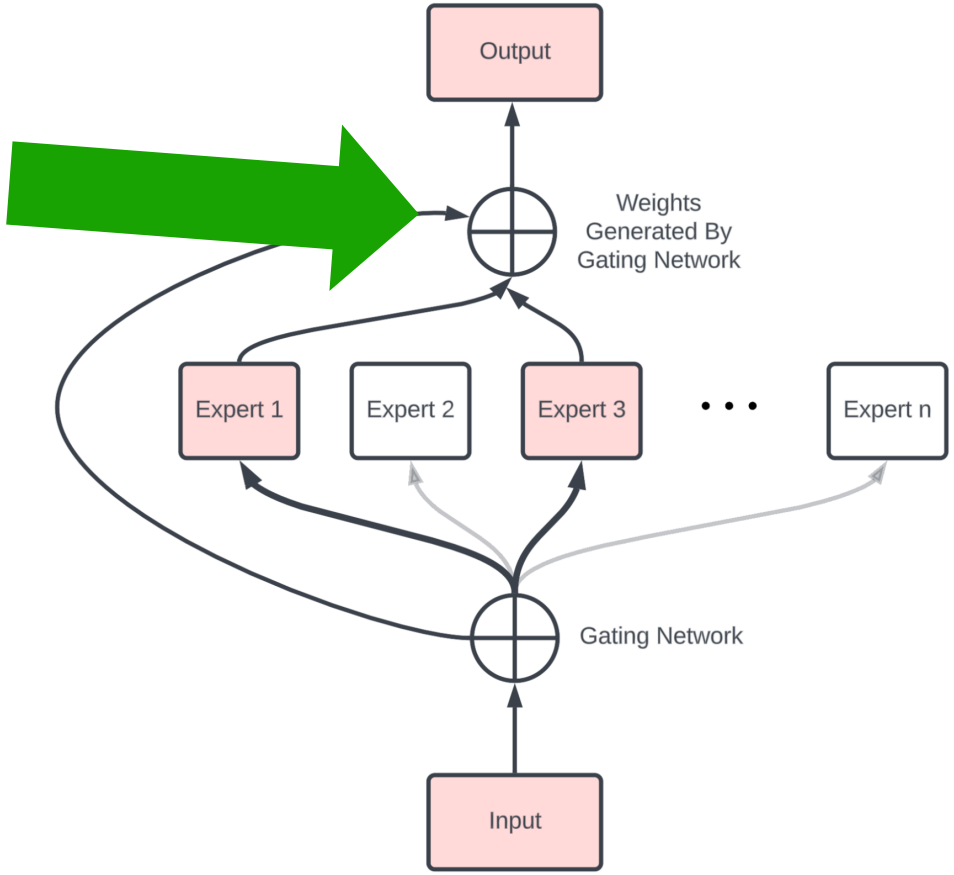
- Se utiliza **un modelo** para interpretar las predicciones hechas por cada experto y para ayudar a **decidir** en qué experto confiar para una determinada información.



Mezcla de expertos

3. Modelo de compuerta (Gating model)

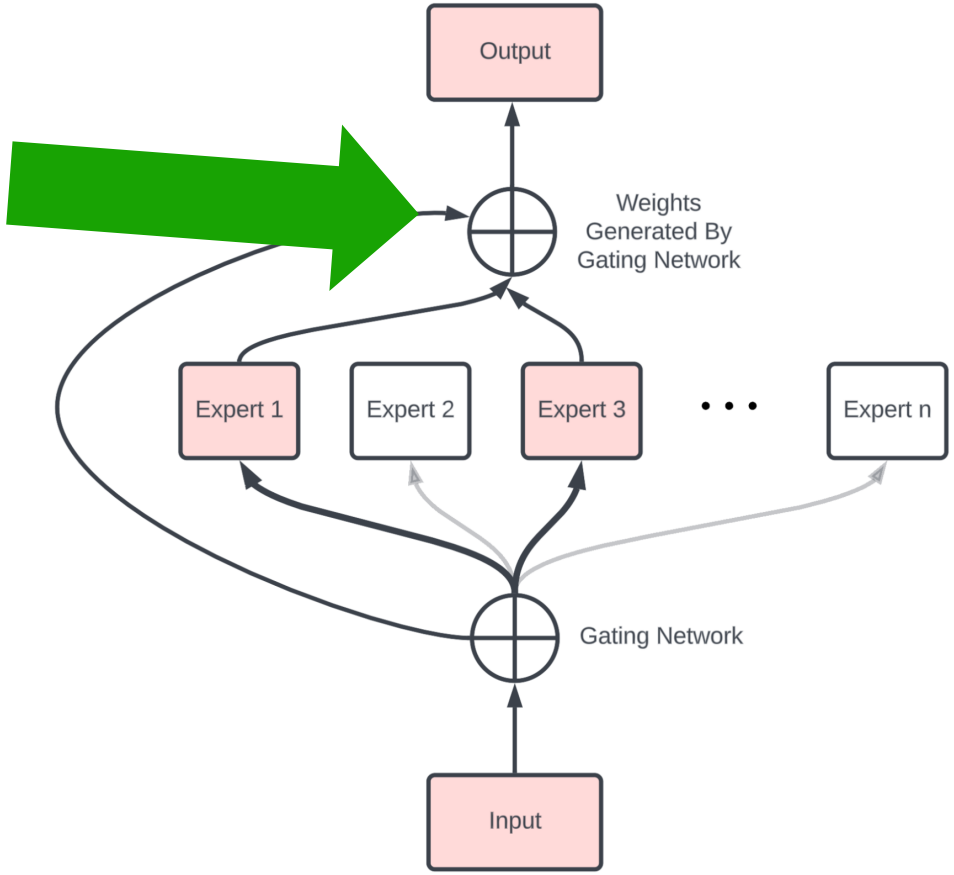
- Se utiliza **un modelo** para interpretar las predicciones hechas por cada experto y para ayudar a **decidir** en qué experto confiar para una determinada información.
- La red de compuerta es clave para el enfoque y, efectivamente, el modelo aprende a elegir el tipo de subtarea para una entrada dada y, a su vez, el experto en el que confiar para hacer una predicción sólida



Mezcla de expertos

3. Modelo de compuerta (Gating model)

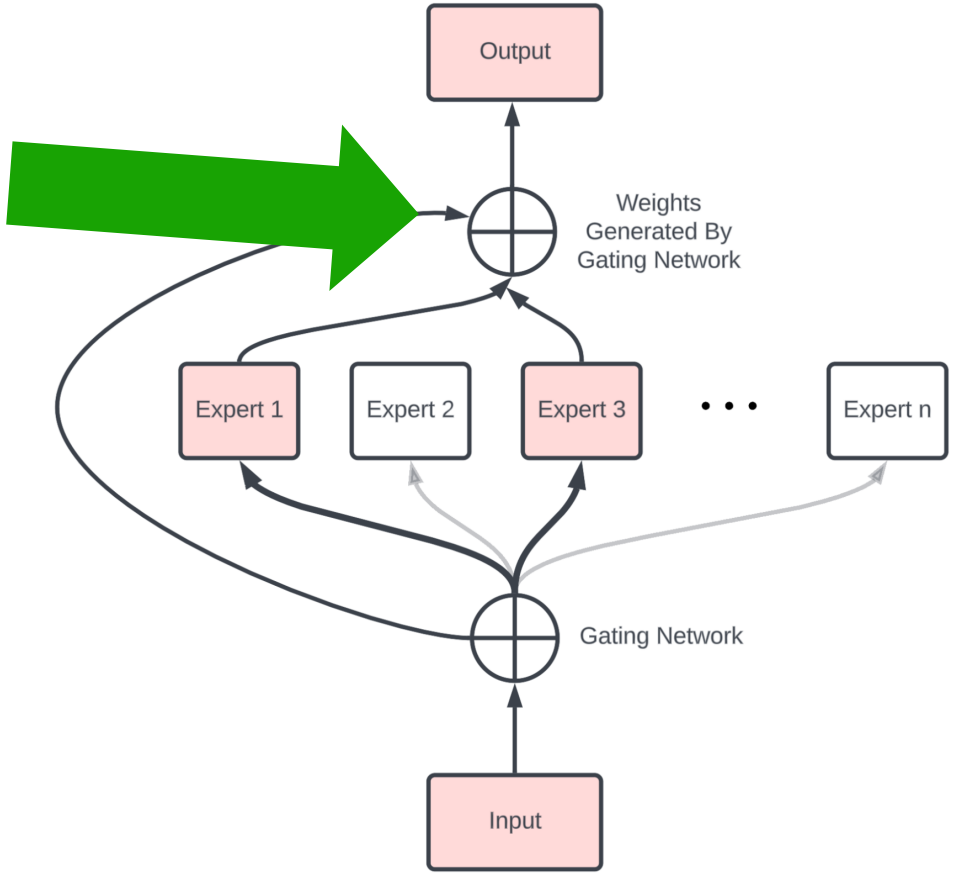
- Se utiliza **un modelo** para interpretar las predicciones hechas por cada experto y para ayudar a **decidir** en qué experto confiar para una determinada información.
- La red de compuerta es clave para el enfoque y, efectivamente, el modelo aprende a elegir el tipo de subtarea para una entrada dada y, a su vez, el experto en el que confiar para hacer una predicción sólida
- Este procedimiento de capacitación se implementó tradicionalmente utilizando la *Expectation Maximization* (*EM*).



Mezcla de expertos

3. Modelo de compuerta (Gating model)

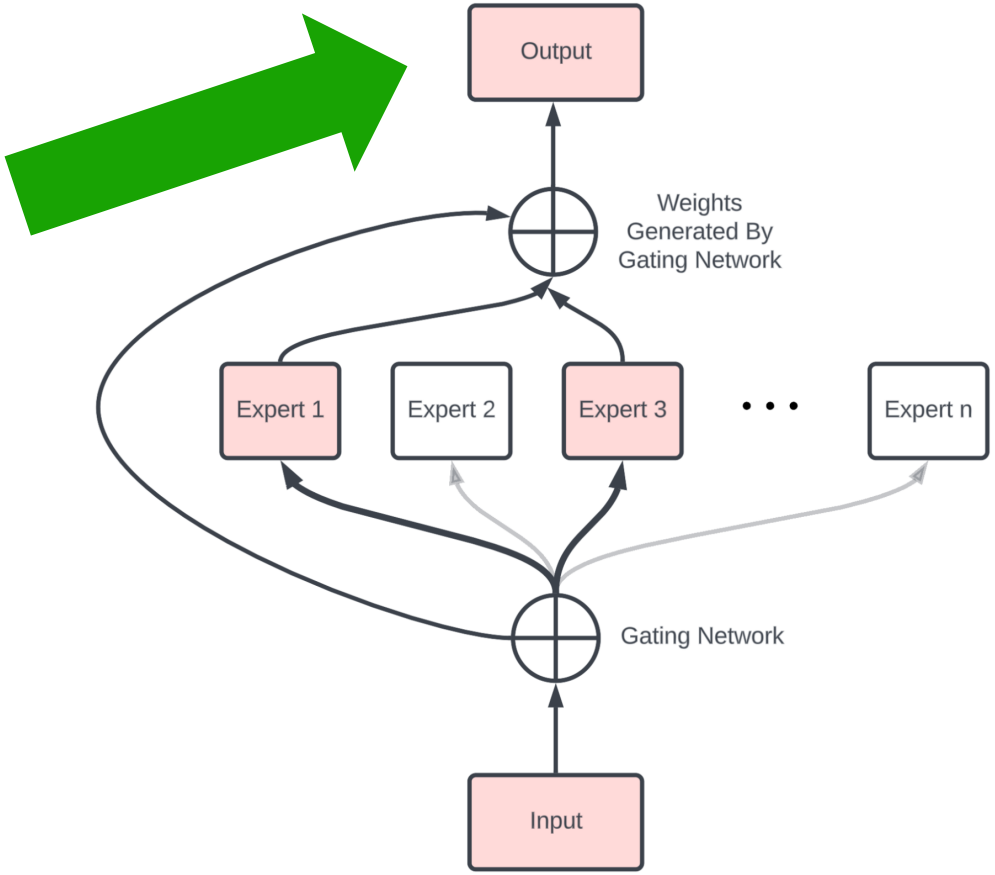
- Se utiliza **un modelo** para interpretar las predicciones hechas por cada experto y para ayudar a **decidir** en qué experto confiar para una determinada información.
- La red de compuerta es clave para el enfoque y, efectivamente, el modelo aprende a elegir el tipo de subtarea para una entrada dada y, a su vez, el experto en el que confiar para hacer una predicción sólida
- Este procedimiento de capacitación se implementó tradicionalmente utilizando la *Expectation Maximization* (*EM*).
- La red de activación puede tener una salida **softmax** que proporcione una puntuación de confianza similar a la probabilidad para cada experto.



Mezcla de expertos

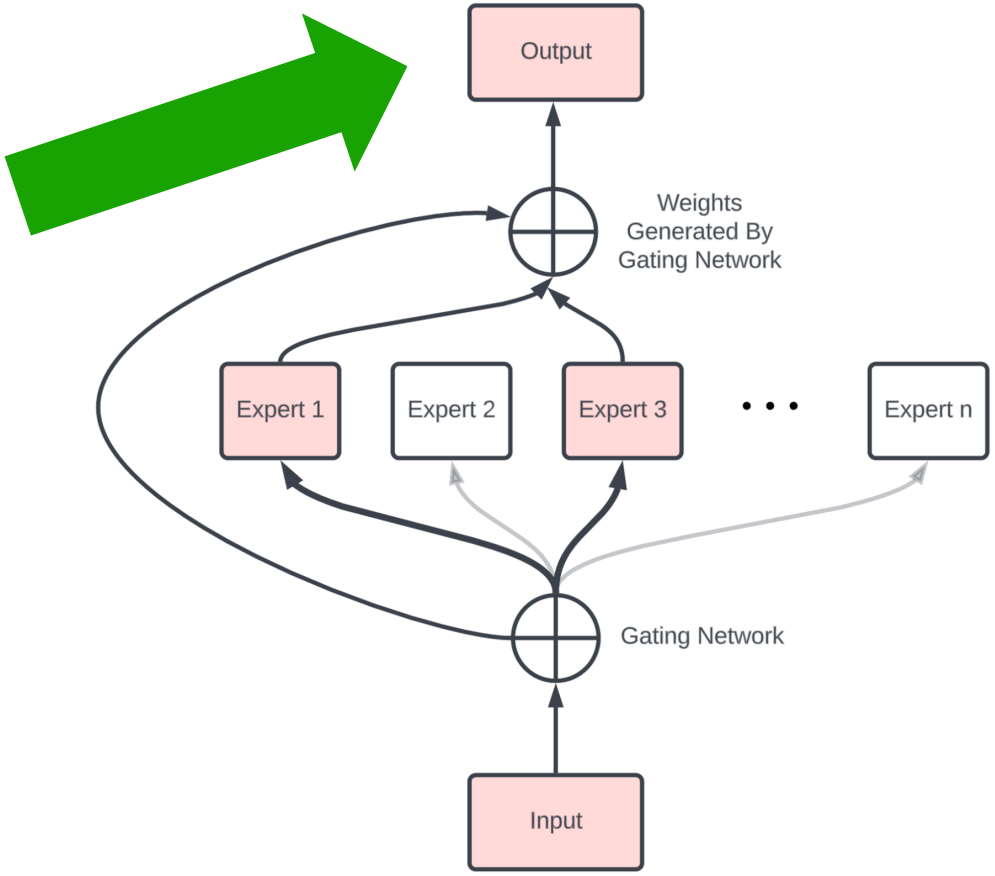
4. Método de agrupación

- Finalmente, la mezcla de modelos expertos debe realizar una **predicción**; esto se logra mediante un mecanismo de **agregación** (pooling).



Mezcla de expertos
4. Método de agrupación

- Finalmente, la mezcla de modelos expertos debe realizar una **predicción**; esto se logra mediante un mecanismo de **agregación** (pooling).
- Esto podría ser tan simple como seleccionar al experto con el **mayor rendimiento** o **confianza** que brinda la red de control.
  - Alternativamente, se podría hacer una predicción de suma ponderada que combine explícitamente las predicciones realizadas por cada experto y la confianza estimada por la red de activación.
  - Existen otros enfoques para hacer un uso eficaz de las predicciones y la salida de la red de control.

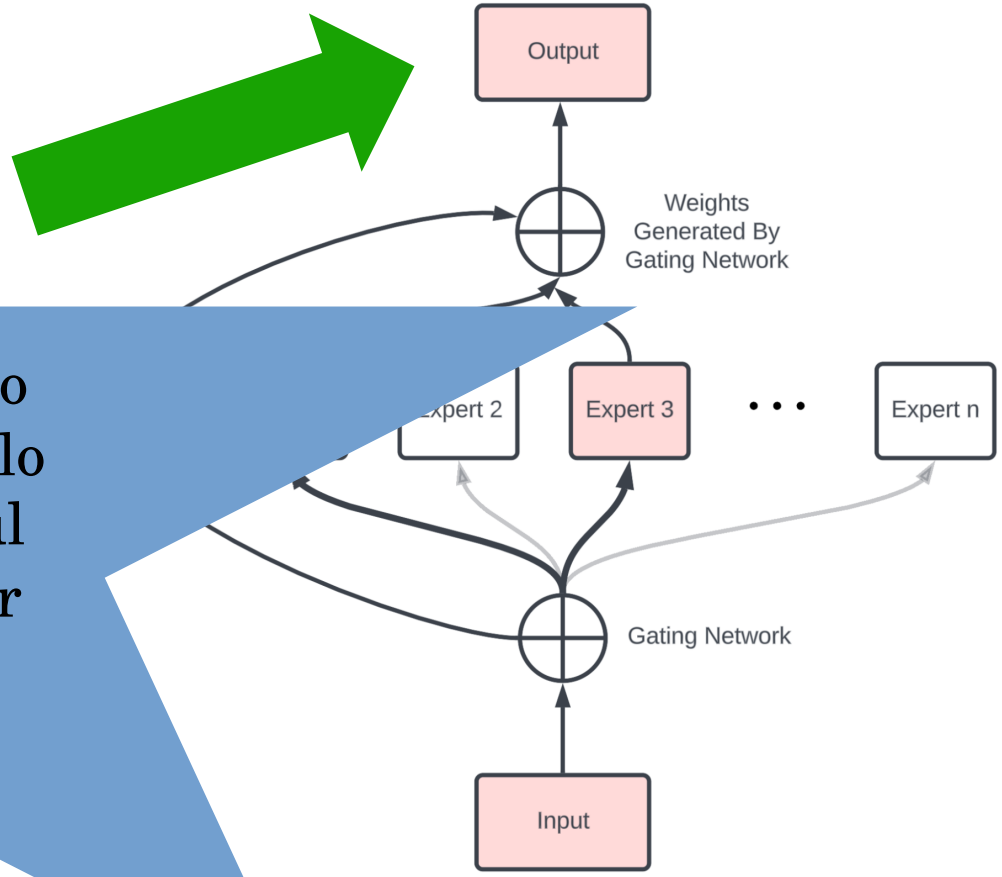


Mezcla de expertos

4. Método de agrupación

- Finalmente, la mezcla de modelos expe ebe realizar una **predicción**; esto se logr te un mecanismo de **agregación** (poolin
- Esto podría ser tan simple como experto que brinda la
  - Alternativamen predicción de suma p explícitamente las predic por cada experto y la co la red de activación.
  - Existen otros enfoqu un uso eficaz de las predico salida de la red de control.

Enfoque en desuso aunque ChatGPT lo ha sacado del baúl ¡nuevamente! (ver foro)



¡Gracias!

Manuel Castillo-Cara, Luis Sarro
www.manuelcastillo.eu
Departamento de Inteligencia Artificial
Escuela Técnica Superior de Ingeniería Informática
Universidad Nacional de Educación a Distancia (UNED)