Análisis exploratorio de los datos

Deep Learning y Series de tiempo

Marco Teran
Universidad Sergio Arboleda

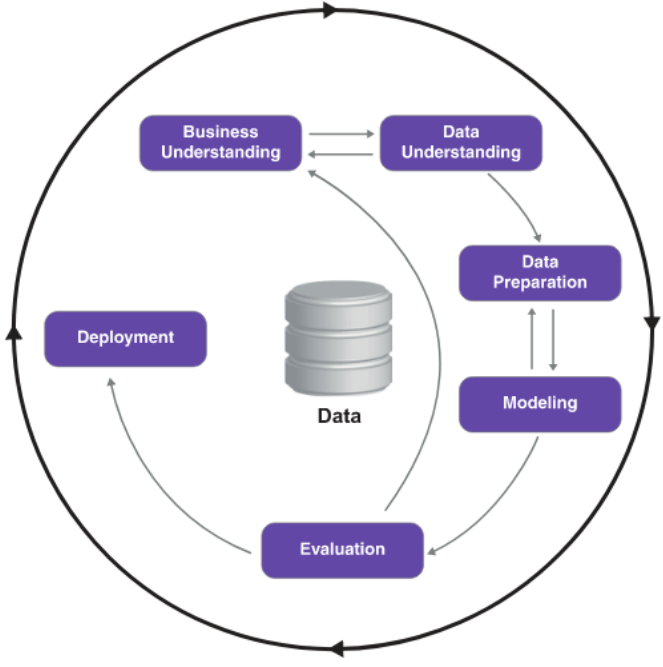
2023

Contenido

Introducción

CRISP-DM

- Crear un sistema de Machine learning implica algo más que seleccionar un modelo, entrenarlo y aplicarlo a nuevos datos.
- Existen marcos que nos ayudan a organizar los proyectos de Machine learning.
- Uno de ellos es CRISP-DM, el Proceso Estándar Intersectorial para la Minería de Datos (1996)


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
Data

CRISP-DM

- **Comprensión del negocio**
  - Identificación del problema
  - Evaluación de si el Machine learning es una herramienta útil
- **Comprensión de los datos**
  - Análisis de los conjuntos de datos disponibles
  - Decisión de si se necesitan recopilar más datos
- **Preparación de datos**
  - Transformación de los datos en forma tabular
- **Modelado**
  - Entrenamiento de un modelo
- **Evaluación**
  - Evaluación del modelo para ver si resuelve el problema original
  - Medición del éxito del modelo
- **Despliegue**
  - Implantación del modelo en el entorno de producción

Ejemplo

Supongamos que queremos construir un sistema de detección de spam: para cada correo electrónico que recibimos, queremos determinar si es spam o no. Si lo es, queremos meterlo en la carpeta de "spam".


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment

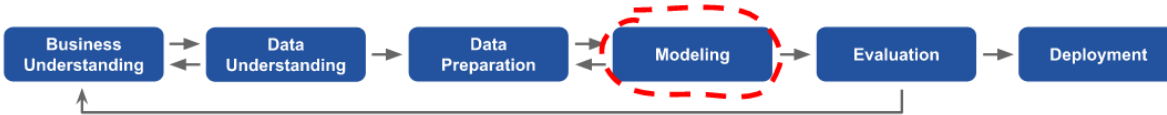
Etapa de comprensión del negocio

- Análisis del problema de mensajes de spam debido a quejas de los usuarios
- Paso de comprensión del negocio para determinar si el Machine learning puede solucionar el problema
  - Análisis de la solución existente
  - Definición del objetivo y de la forma de medirlo
  - Ejemplos de objetivos: "Reducir el número de mensajes de spam denunciados" o "Reducir el número de quejas sobre spam que el servicio de atención al cliente recibe al día"
  - Propuesta de una solución alternativa si el Machine learning no es adecuado


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment

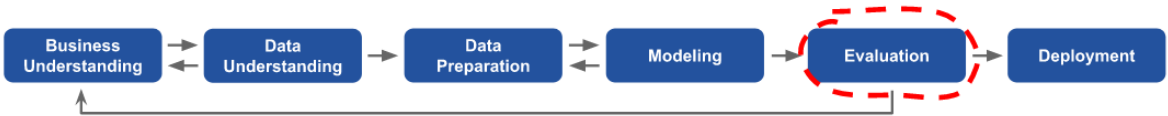
Paso de comprensión de datos

- Paso de comprensión de los datos para identificar las fuentes de datos
  - Ejemplo de fuente de datos: botón de Informar de Spam
  - Análisis y examen de los datos para determinar si son adecuados para resolver el problema
  - Razones por las que los datos pueden no ser adecuados: tamaño insuficiente del conjunto de datos, ruido excesivo, procesos de recopilación de datos rotos, entre otros
  - Necesidad de obtener mejores datos mediante la adquisición de fuentes externas o la mejora de los procesos de recopilación internos
  - Posible impacto de los hallazgos en el objetivo establecido en el paso de comprensión del negocio, lo que podría requerir ajustes en el objetivo
- Fase de preparación de datos, una vez que dispongamos de fuentes de datos fiables


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment

Paso de modelado

- Paso de comprensión de los datos para identificar las fuentes de datos
  - Selección del modelo de Machine learning y definición de cómo medir su rendimiento.
  - Posibilidad de volver atrás y ajustar la forma en que se prepararon los datos.
  - Necesidad de establecer un marco de validación adecuado.
  - Selección del mejor modelo posible y paso a la fase de evaluación.


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment

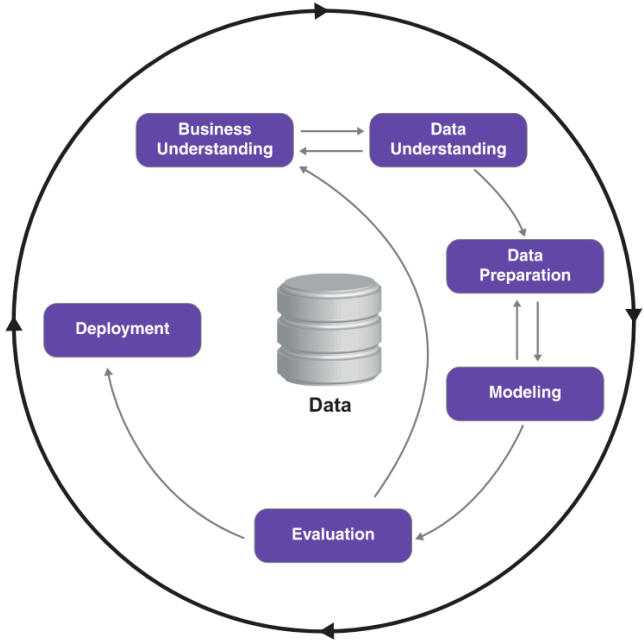
Etapa de evaluación

- Comprobar si el modelo cumple las expectativas
- Fijar el objetivo y métrica empresarial en el paso de comprensión del negocio
- Asegurarse de que el modelo mueve la métrica en la dirección correcta
- La métrica puede ser el número de personas que hacen clic en el botón Informar de spam o el número de quejas recibidas por el servicio de atención al cliente
- Esperar que el uso del modelo reduzca el número de métricas identificadas.


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment

Etapa de despliegue

- La mejor forma de evaluar un modelo es desplegarlo en una fracción de usuarios y comprobar si nuestra métrica de negocio cambia para estos usuarios.
- Si queremos que nuestro modelo reduzca el número de mensajes de spam notificados, esperamos ver menos informes en este grupo en comparación con el resto de los usuarios.
- Utilizamos todo lo aprendido en todos los pasos y volvemos al primero para reflexionar sobre lo que hemos conseguido.
- Podemos darse cuenta de que nuestro objetivo inicial era erróneo y que lo que en realidad queremos hacer no es reducir el número de informes, sino aumentar el compromiso de los clientes disminuyendo la cantidad de spam.
- Volvemos al paso de comprensión del negocio para redefinir nuestro objetivo.
- Al evaluar el modelo de nuevo, utilizaremos una métrica de negocio diferente para medir su éxito.


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
Data

Iterar

- CRISP-DM enfatiza la naturaleza iterativa del proceso de Machine learning.
- Siempre se espera que se vuelva al primer paso después del último paso para refinar el problema original y cambiarlo en base a la información aprendida.
- No se detienen en el último paso, sino que se replantean el problema para mejorar en la siguiente iteración.
- Es un error común pensar que los ingenieros de Machine learning y los científicos de datos se pasan todo el día entrenando modelos.
- El diagrama de CRISP-DM muestra que hay muchos pasos antes y después del modelado que son importantes para el éxito de un proyecto de Machine learning.

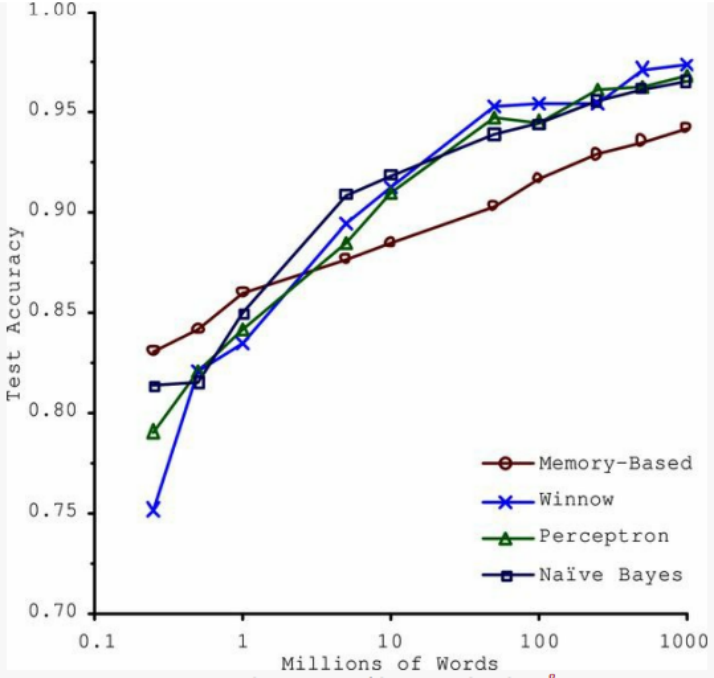
Principales desafíos del Machine Learning

Principales desafíos del Machine learning

Dado que su tarea principal es seleccionar un modelo y entrenarlo con algunos datos, las dos cosas que pueden salir mal son un "modelo deficiente" y "datos deficientes".

Cantidad insuficiente de datos de entrenamiento

- El Machine learning aún no ha llegado a ese punto.
- La mayoría de los algoritmos de Machine learning requieren mucha data para funcionar correctamente.
- Incluso para problemas muy simples, normalmente necesitas miles de ejemplos, y para problemas complejos como el reconocimiento de imágenes o voz, puede ser necesario tener millones de ejemplos (a menos que se puedan reutilizar partes de un modelo existente).


1.00
0.95
0.90
0.85
0.80
0.75
0.70
Test Accuracy
0.1
1
10
100
1000
Millions of Words
Memory-Based
Winnow
Perceptron
Naïve Bayes

Datos de formación no representativos

Para generalizar bien, es crucial que los datos de entrenamiento sean representativos de los nuevos casos que deseas generalizar. Esto es cierto tanto para el aprendizaje basado en instancias como para el basado en modelos.

- Si los datos de entrenamiento no son representativos, el modelo resultante no será preciso.
- Es necesario asegurarse de que la muestra sea representativa: esto puede ser difícil de lograr ya que incluso muestras grandes pueden ser no representativas si el método de muestreo es defectuoso. Esto se llama sesgo de muestreo.

Datos de baja calidad

- Errores, valores atípicos y ruido en los datos de entrenamiento dificultan que el sistema detecte patrones subyacentes y funcione bien.
- Es importante limpiar los datos de entrenamiento y muchos científicos de datos dedican tiempo a hacerlo.
- Algunos casos donde conviene limpiar los datos de entrenamiento son:
  - Si hay instancias claramente atípicas, se pueden descartar o corregir manualmente.
  - Si faltan características en algunas instancias, se puede ignorar el atributo, ignorar las instancias, rellenar los valores faltantes o entrenar un modelo con la característica y otro sin ella.

Características irrelevantes

- El éxito de un proyecto de Machine learning depende de contar con un conjunto de características relevantes y no irrelevantes.
- La ingeniería de características es el proceso que implica seleccionar y extraer características útiles para el entrenamiento.
- Los pasos de la ingeniería de características son:
  - **Selección de características** (elegir las más útiles de entre las existentes).
  - **Extracción de características** (combinar las existentes para obtener una más útil).
- Los algoritmos de reducción de dimensionalidad pueden ayudar en la extracción de características.
- Es importante destacar que existen algoritmos malos en el Machine learning.

Sobreajuste (overfitting) y Regularización

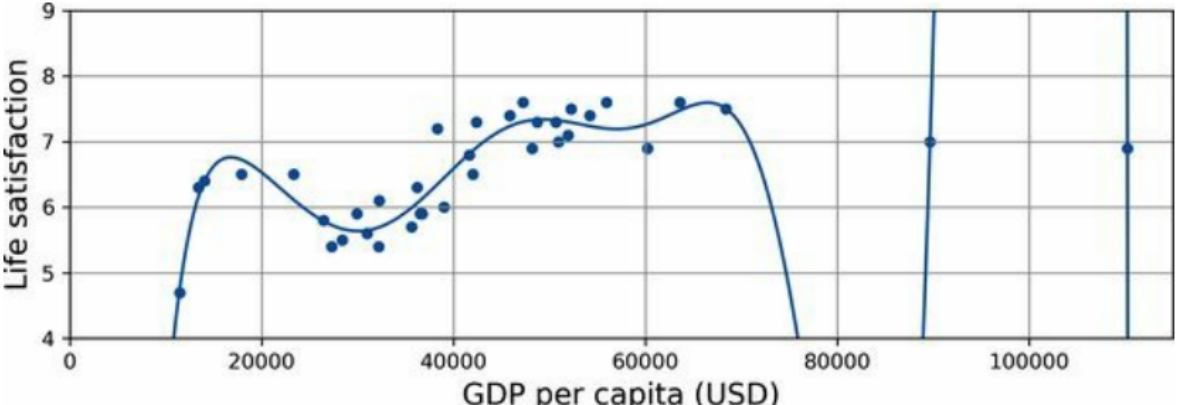
- Supongamos que visita un país extranjero y el taxista le estafa. Podrías tener la tentación de decir que todos los taxistas de ese país son ladrones.
  - Generalizar en exceso es algo que los humanos hacemos con demasiada frecuencia y, por desgracia, las máquinas pueden caer en la misma trampa si no tenemos cuidado.

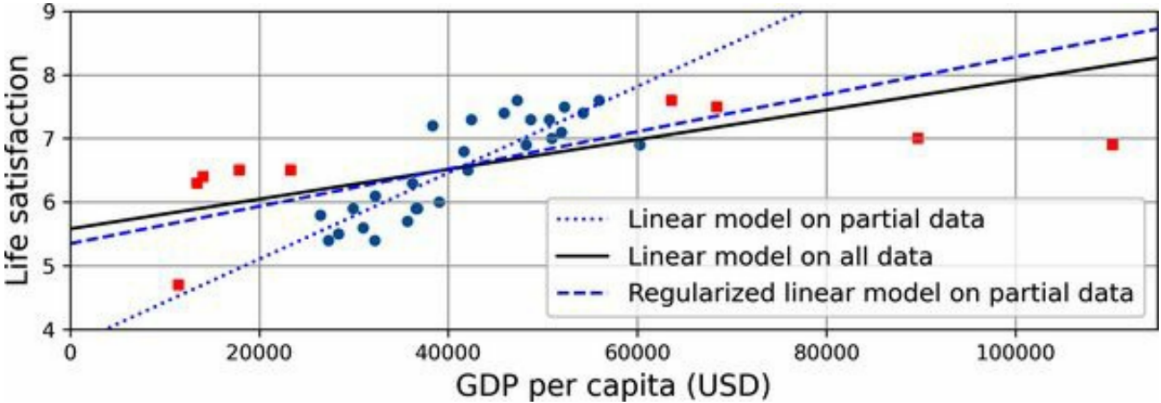
Sobreajuste y Regularización

- En el Machine learning, esto se llama sobreajuste: significa que el modelo funciona bien con los datos de entrenamiento, pero no generaliza bien.
- Los modelos complejos, como las redes neuronales profundas, pueden detectar patrones sutiles en los datos, pero si el conjunto de entrenamiento es ruidoso, o si es demasiado pequeño, lo que introduce ruido de muestreo, es probable que el modelo detecte patrones en el propio ruido.
  - Los patrones detectados pueden ser resultado del ruido en los datos, por lo que no se generalizarán a nuevos casos.

Sobreajuste y Regularización

- Soluciones al sobreajuste:
  - Simplificar el modelo seleccionando uno con menos parámetros, reduciendo el número de atributos en los datos de entrenamiento o restringiendo el modelo.
  - Recopilar más datos de entrenamiento.
  - Reducir el ruido en los datos de entrenamiento corrigiendo errores y eliminando valores atípicos.
  - La restricción de un modelo para simplificarlo y reducir el riesgo de sobreajuste se denomina regularización.
- La cantidad de regularización que se aplica durante el aprendizaje puede controlarse mediante un hiperparámetro.


Life satisfaction
9
8
7
6
5
4
0
20000
40000
60000
80000
100000
GDP per capita (USD)


Life satisfaction
GDP per capita (USD)
Linear model on partial data
Linear model on all data
Regularized linear model on partial data
9
8
7
6
5
4
0
20000
40000
60000
80000
100000

Ajuste insuficiente (underfitting) de los datos de entrenamiento

- El underfitting es lo contrario del sobreajuste.
- Se produce cuando el modelo es demasiado simple para aprender la estructura subyacente de los datos.
- Un modelo lineal de satisfacción vital es propenso al infraajuste.
- Las predicciones serán inexactas, incluso en los ejemplos de entrenamiento.

Ajuste insuficiente (underfitting) de los datos de entrenamiento

- Las principales opciones para solucionar este problema son:
  - Seleccionar un modelo más potente, con más parámetros.
  - Introducir mejores características en el algoritmo de aprendizaje (ingeniería de características).
  - Reducir las restricciones del modelo (por ejemplo, reduciendo el hiperparámetro de regularización).

Resumen

- El Machine learning consiste en hacer que las máquinas mejoren en alguna tarea mediante aprender de los datos, en lugar de tener que codificar reglas explícitamente.
- Hay muchos tipos diferentes de sistemas de Machine learning: supervisados o no, por lotes o en línea, basados en instancias o en modelos.
- En un proyecto de ML se reúnen datos en un conjunto de entrenamiento y se alimenta el conjunto de entrenamiento a un algoritmo de aprendizaje.
- Si el algoritmo se basa en un modelo, ajusta algunos parámetros para adaptar el modelo al conjunto de entrenamiento y luego, con suerte, podrá hacer buenas predicciones en nuevos casos.

Resumen

- El Machine learning consiste en hacer que las máquinas mejoren en alguna tarea mediante aprender de los datos, en lugar de tener que codificar reglas explícitamente.
- Hay muchos tipos diferentes de sistemas de Machine learning: supervisados o no, por lotes o en línea, basados en instancias o en modelos.
- En un proyecto de ML se reúnen datos en un conjunto de entrenamiento y se alimenta el conjunto de entrenamiento a un algoritmo de aprendizaje.
- Si el algoritmo se basa en un modelo, ajusta algunos parámetros para adaptar el modelo al conjunto de entrenamiento y luego, con suerte, podrá hacer buenas predicciones en nuevos casos.

Pruebas y validación

Pruebas y validación

- La mejor forma de saber si un modelo se generaliza es probándolo con nuevos casos.
- Dividir los datos en dos conjuntos: entrenamiento y prueba.
- La tasa de error en los nuevos casos se llama error de generalización.
- El error de generalización se estima evaluando el modelo en el conjunto de prueba.
- Si el error de entrenamiento es bajo pero el error de generalización es alto, significa que el modelo se está ajustando en exceso a los datos de entrenamiento.
- El tamaño del conjunto de prueba depende del tamaño del conjunto de datos.

Ajuste de hiperparámetros y selección de modelos

Ajuste de hiperparámetros y selección de modelos

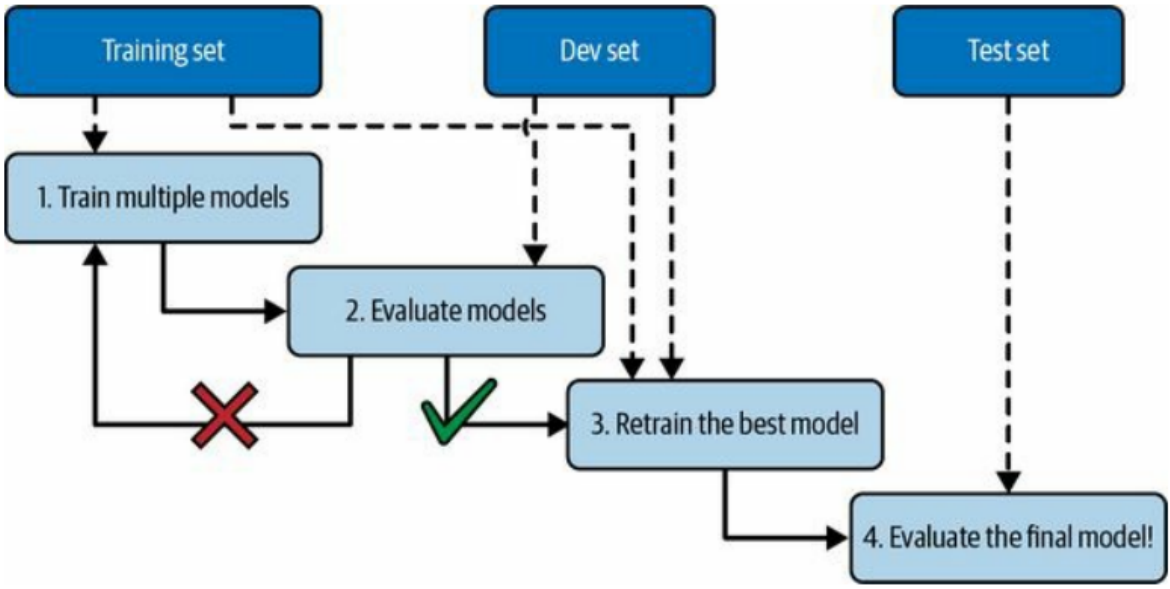
- Para decidir entre dos modelos se pueden comparar sus grados de generalización con un conjunto de pruebas.
- La regularización se aplica para evitar el sobreajuste en un modelo que generaliza mejor.
- Una forma de elegir el valor del hiperparámetro de regularización es entrenar varios modelos con distintos valores del hiperparámetro.
- La validación por exclusión es una solución común para el problema de adaptación de modelos y hiperparámetros al conjunto de pruebas.
- La validación por exclusión implica excluir parte del conjunto de entrenamiento para evaluar varios modelos y seleccionar el mejor.

Ajuste de hiperparámetros y selección de modelos

- El conjunto retenido se denomina conjunto de validación o conjunto de desarrollo.
- Después del proceso de validación, se entrena el mejor modelo en el conjunto de entrenamiento completo.
- Por último, se evalúa el modelo final en el conjunto de pruebas para obtener una estimación del error de generalización.
- Si el conjunto de validación es demasiado pequeño, las evaluaciones del modelo serán imprecisas.

Ajuste de hiperparámetros y selección de modelos

- Si el conjunto de validación es demasiado grande, el conjunto de entrenamiento restante será mucho más pequeño que el conjunto de entrenamiento completo.
- Una forma de resolver este problema es realizar una validación cruzada repetida, utilizando muchos conjuntos de validación pequeños.
- El tiempo de entrenamiento se multiplica por el número de conjuntos de validación en la validación cruzada repetida.


Training set
Dev set
Test set
1. Train multiple models
2. Evaluate models
3. Retrain the best model
4. Evaluate the final model!

Divergencia de datos

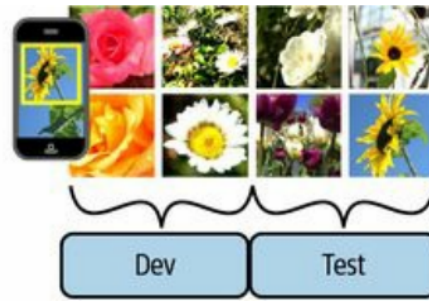
Divergencia de datos

- Los datos utilizados en el entrenamiento pueden no ser perfectamente representativos de los datos de producción.
- Los conjuntos de validación y prueba deben ser representativos de los datos de producción.
- Se recomienda usar imágenes representativas y sin duplicados para los conjuntos de validación y prueba.
- Si el rendimiento del modelo en el conjunto de validación es bajo, puede deberse al sobreajuste o la falta de correspondencia entre los datos de entrenamiento y los de producción.
- Una solución es mantener un conjunto de entrenamiento de desarrollo para evaluar el modelo después del entrenamiento.

Divergencia de datos

- Si el modelo funciona mal en el conjunto de entrenamiento y desarrollo, puede intentar simplificar o regularizar el modelo, obtener más datos de entrenamiento y limpiar los datos existentes.
- Si el modelo funciona bien en el conjunto de entrenamiento y desarrollo, se puede evaluar en el conjunto de desarrollo.
- Si el rendimiento en el conjunto de desarrollo es bajo, puede preprocesar los datos para que se parezcan más a los de producción y volver a entrenar el modelo.
- Una vez que se tenga un modelo que funcione bien en el conjunto de entrenamiento y desarrollo, se puede evaluar en el conjunto de prueba.
- El conjunto de prueba proporciona información sobre cómo funcionará el modelo en producción.


Train
Train-dev


Dev
Test

Proyecto de Machine Learning de principio a fin

Exploración de datos

Exploración de datos

La exploración de datos en Machine Learning se refiere al proceso de analizar y visualizar los datos disponibles antes de aplicar técnicas de Machine learning. El objetivo es comprender las características de los datos y descubrir patrones o relaciones que puedan ser útiles para el modelo de Machine learning.

Principales características de la exploración de datos

1. Revisión y limpieza de datos
2. Identificación de variables relevantes
3. Análisis de distribuciones y patrones de datos
4. Visualización de datos para detectar tendencias
5. Identificación de valores atípicos y datos faltantes.

Principales características de la exploración de datos

1. Revisión y limpieza de datos
2. Identificación de variables relevantes
3. Análisis de distribuciones y patrones de datos
4. Visualización de datos para detectar tendencias
5. Identificación de valores atípicos y datos faltantes.

Ventajas de la exploración de datos

1. Permite identificar problemas en los datos y corregirlos antes de entrenar el modelo
2. Ayuda a seleccionar variables importantes para el modelo
3. Permite descubrir patrones y relaciones que pueden ser útiles para la predicción
4. Ayuda a evitar el sobreajuste del modelo
5. Facilita la comprensión de los datos y la interpretación de los resultados del modelo.

Técnicas de exploración de datos

1. Análisis estadístico descriptivo
2. Gráficos de dispersión, histogramas y diagramas de cajas
3. Correlación y análisis de regresión
4. Visualización de datos en mapas y gráficos de red
5. Análisis de componentes principales y técnicas de reducción de la dimensionalidad.

Pasos para resolver un problema mediante la exploración de los datos

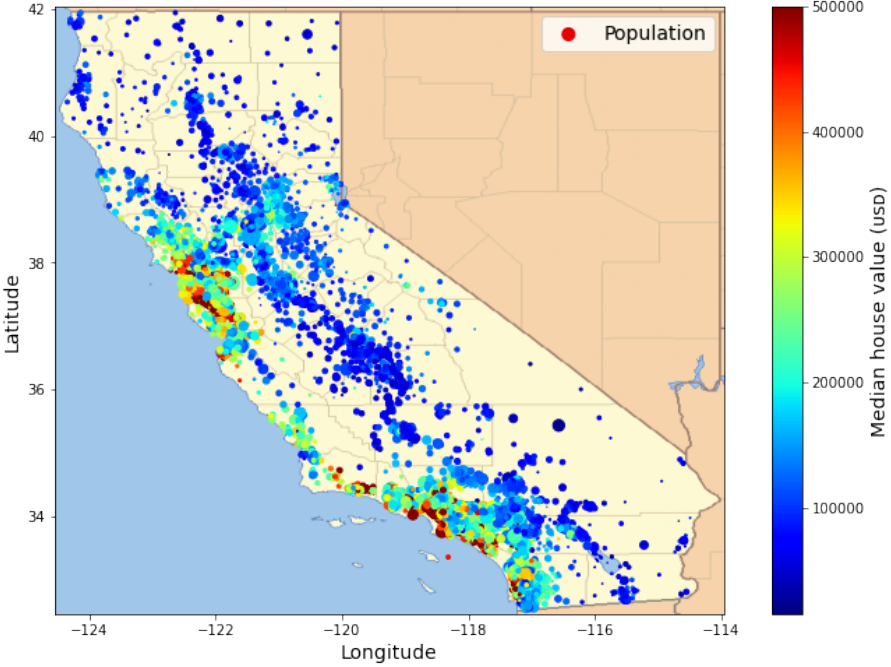
Aquí están los pasos principales que recorreremos:

1. Analizar el panorama general.
2. Obtener los datos.
3. Explorar y visualizar los datos para obtener información.
4. Preparar los datos para los algoritmos de machine learning.

Dataset de los precios de las casas de California

Aquí están los pasos principales que recorreremos:

1. Utilizaremos el conjunto de datos de los precios de las casas de California (California Housing Prices).
2. Este conjunto de datos está disponible en el repositorio StatLib2 y se basa en los datos del censo de California de 1990.
3. Aunque no es reciente, tiene muchas cualidades para el aprendizaje y lo utilizaremos como si fueran datos actuales.
4. Se ha añadido un atributo categórico y se han eliminado algunas características con fines didácticos.


Population
Latitude
Longitude
42
40
38
36
34
−124
−122
−120
−118
−116
−114
Median house value (usd)
500000
400000
300000
200000
100000

Dataset de los precios de las casas de California

- El conjunto de datos California Housing Prices contiene información sobre precios de viviendas y características de diferentes regiones de California.
- Contiene 20,640 instancias con 8 atributos para cada una, incluyendo la longitud y latitud de la región, la edad media de las viviendas, el número medio de habitaciones, el número medio de habitantes por hogar, el ingreso medio de los hogares en la región, la población total de la región y el valor medio de la vivienda.
- Los valores de los atributos numéricos varían en diferentes rangos, lo que puede requerir una normalización antes de aplicar ciertos algoritmos de Machine learning.
- El conjunto de datos se puede utilizar para diferentes tareas, como la predicción del valor medio de la vivienda en una región, la identificación de patrones geográficos en los precios de las viviendas y la exploración de relaciones entre los diferentes atributos.

Tener una visión amplia

- La tarea es crear un modelo de precios de casas en California utilizando el censo de California.
- Los datos incluyen métricas como la población, los ingresos medios y los precios medios de las casas para cada distrito en California.
- Los distritos son la unidad geográfica más pequeña para la que la Oficina del censo de EE. UU. publica datos simples, con una población de entre 600 y 3,000 personas.
- El modelo debe aprender a partir de estos datos y ser capaz de predecir el precio medio de las casas en cualquier distrito si se le dan todas las demás métricas.

Tener una visión amplia

- Se recomienda el uso de una lista de comprobación de proyectos de machine learning como herramienta de organización.
- Se puede utilizar la lista de comprobación de proyectos de machine learning del apéndice B, pero se debe adaptar a las necesidades específicas del proyecto.
- En este capítulo se cubrirán muchos puntos de la lista, pero se omitirán algunos porque son evidentes o se tratarán en capítulos posteriores.

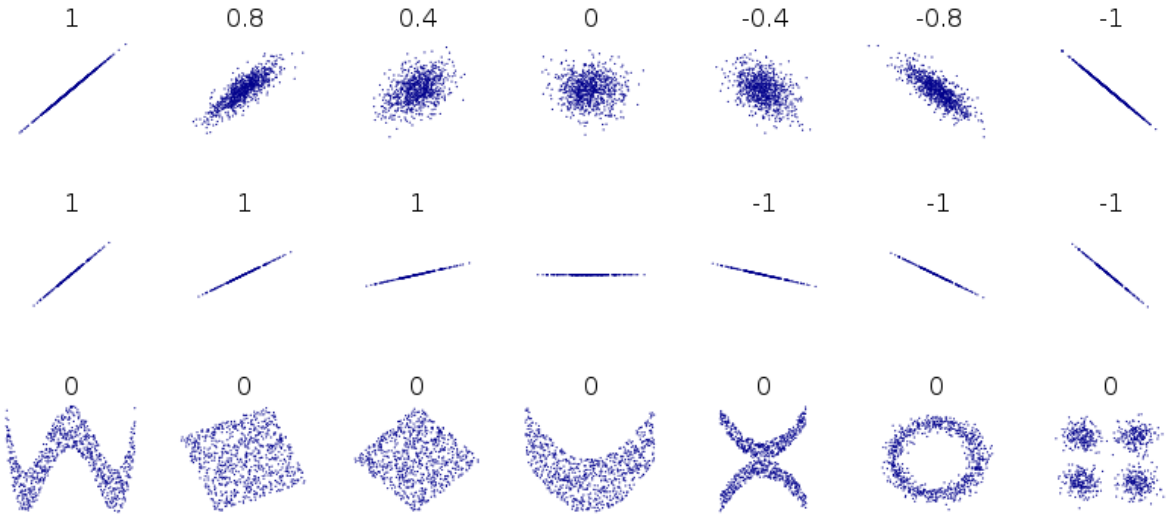
Enmarcar el problema

- Es importante preguntar al jefe cuál es el objetivo empresarial y cómo se espera utilizar el modelo para beneficiarse de él.
- El conocimiento del objetivo determinará la forma en que se enmarcará el problema, qué algoritmos seleccionar, qué medida de rendimiento utilizar para evaluar el modelo y cuánto esfuerzo se dedicará a ajustarlo.
- La respuesta del jefe indica que la salida del modelo se integrará en otro sistema de machine learning, el cual determinará si es rentable invertir en una determinada área o no.
- La precisión de este sistema es fundamental, ya que afecta directamente a los ingresos.

Pipelines

Una secuencia de datos que procesan componentes se llama pipeline de datos. Las pipelines son muy comunes en los sistema de machine learning, puesto que hay muchos datos que manipular y muchas transformaciones de datos que aplicar.

- Los componentes se ejecutan de forma asíncrona, procesando datos y almacenando resultados para que los utilicen otros componentes.
- Los componentes son independientes y se comunican a través del almacenamiento de datos compartido, lo que hace que el sistema sea fácil de entender y que diferentes equipos puedan trabajar en diferentes componentes.
- Si un componente falla, los componentes aguas abajo pueden continuar usando el último resultado, lo que hace que la arquitectura sea robusta, pero se necesita un monitoreo adecuado para evitar datos obsoletos.
- Los datos obsoletos pueden disminuir el rendimiento general del sistema si un componente defectuoso pasa desapercibido.


1
0.8
0.4
0
-0.4
-0.8
-1
1
1
1
-1
-1
-1
0
0
0
0
0
0
0

¡Muchas gracias por su atención!

¿Preguntas?

Contacto: Marco Teran
webpage: marcoteran.github.io/
e-mail: marco.teran@usa.edu.co

UNIVERSIDAD
SERGIO ARBOLEDA