

# Resumen y Revisión Fin de Semana I

## Ciclo de vida de un modelo de ML

## Problema de Negocio

- Reducir coste
- Mejorar margen
- Incrementar satisfacción de cliente
- Métricas de éxito

#### Trabajo de Limpieza

- Buscar nuevas fuentes de datos relevantes
- Calidad de los datos
- Explorar, visualizar
- Ingeniería de características

#### Creación del modelo

- Seleccionar el algoritmo adecuado
- Parametrizar el modelo

#### Evaluar y Optimizar

- Hasta que encontremos los parámetros adecuados
- Evaluar el modelo
- Optimizar parámetros

#### Más validación

- Pruebas en línea
- A/B test

## Tareas de Limpieza de Datos

- Duplicados y Errores
  - Tratarlos
- Nulos
  - Imputarlos → sklearn.impute
    - Univariate → Usa valores solo de esa característica
      - SimpleImputer(missing\_values=np.nan, strategy='mean'
        - » Strategy: mean, Median, most\_frequent, constant
    - Multivariate → Usa valores de todo el conjunto
      - IterativeImputer(missing\_values=0, random\_state=0, n\_nearest\_features=5)
        - » "Nearest" → Índice de correlación
- ¿Cómo marcar?
  - Los imputadores tienen un parámetro add\_indicator
  - Podemos utilizar MissingIndicator para marcar las filas procesadas, o para marcar aquellas en las que tenemos / no tenemos un determinado valor

© 2019 SolidQ

# Tareas de Limpieza (II)

- Escalado
  - Lo vemos en detalle ahora...
- Codificación de valores categóricos
  - from sklearn.preprocessing import LabelEncoder labelencoder\_X = LabelEncoder()
    - Sexo 1, 0..... Seguro???
  - from sklearn.preprocessing import OneHotEncoder
  - enc = OneHotEncoder(handle\_unknown='ignore')
- Tenemos inverse\_Transform!!!

© 2019 SolidQ

# Tipos Aprendizaje

- Aprendizaje Supervisado tiene un conjunto definido de entradas y salidas
  - Datos etiquetados
  - Feedback directo
  - Predice salida / futuro
- Aprendizaje No Supervisado tiene entradas pero las salidas son desconocidas
  - No tenemos etiquetas
  - No feedback
  - Busca estructuras ocultas en los datos
- Aprendizaje Reforzado
  - Proceso de decisión
  - Sistema de recompensa
  - Aprende una serie de acciones

# Separando Conjunto de Datos: Train / Test

### Overfitting

- Hemos entrenado demasiado bien a nuestro modelo
- Se adapta demasiado a los datos de entrenamiento
- Suele ocurrir con conjuntos de datos complejos
  - Muchas características para pocas ocurrencias

#### Underfitting

- No se ajusta a los datos de entrenamiento...no se entera de las tendencias
- Resultado de un modelo con pocas variables independientes
- Cross-validation.... Veremos en un rato

© 2019 SolidQ

# **C** SolidQ