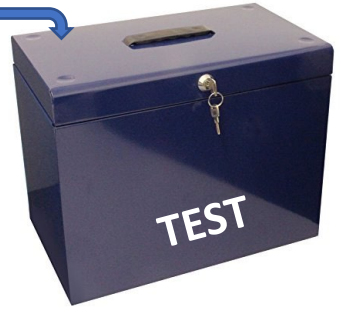
DATASET

Clasificación → Fuga

Regresión → Gasto

Edad	Antigüedad	Género	Localidad	Ingresos	Fuga	Gasto
27	10	H	Madrid	127442	1	120
12	12	M	Sevilla	**NA**	0	25
NA	1	No binario	Bilbao	90233	0	75
160	3	M	Valencia	18923	1	100
…	…	…	…	…	…	…

1. **Dividir en TRAIN y <u>TEST</u>**



2. **Sobre el conjunto de TRAIN:**
   - Computar missings
   - Computar outliers
   - Feature engineering (generar nuevas variables) [opcional]
   - Transformar variables categóricas a numéricas
   - Analizar correlaciones. Eliminar variables muy correlacionadas, descartar variables poco correlacionadas con el target
   - Estandarizar variables si el algoritmo de Machine Learning lo requiere (regularización): *MinMaxScaler*, *StandardScaler*
   - En regresión, transformar target si su distribución no es normal [opcional]
   - En clasificación, balancear clases [opcional]

3. **Entrenar** uno o varios modelos, optimizando hiperparámetros de cada uno (*GridSearchCV, RandomSearchCV, Bayesian*)

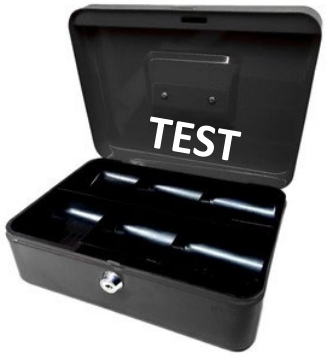
REGRESIÓN:

- Regresión Lineal
- Ridge
- Lasso
- Elastic Net
- Regresión polinómica. Es una regresión lineal con features sintéticas, por ejemplo Edad^2. Si utilizamos esta técnica habría que analizar las variables sintéticas nuevamente (correlaciones, escalado, etc.)
- Decision Trees
- Random Forest
- Boosting (XGBoost, AdaBoost, CatBoost, LightGBM…)

CLASIFICACIÓN

- Logistic Regression
- Decision Trees
- Random Forest
- Boosting (XGBoost, AdaBoost, CatBoost, LightGBM…)

4. **Medir desempeño** en TRAIN y TEST



En el conjunto de TEST tengo que aplicar todas las transformaciones que haya hecho en TRAIN: computar missings, outliers, variables categóricas estandarizar, etc. y después hacer el predict

REGRESIÓN:

- Error cuadrático medio (MSE, RMSE)
- Mean Absolute error (MAE)
- Mean Absolute Percentage error (MAPE)
- Coeficiente de determinación (R^2)
- Correlación entre target y predicción

CLASIFICACIÓN:

- Matriz de confusión
- Accuracy
- Precision
- Recall
- F1 Score
- ROC, AUC

Si el desempeño en TRAIN >> desempeño en TEST → Overfitting

5. **Escoger el modelo (y la estrategia) con mejor desempeño en TEST**

Al final, lo que producimos no son tanto modelos sino **PIPELINES**

EJEMPLO