ITBA
Instituto Tecnológicc
de Buenos Aires

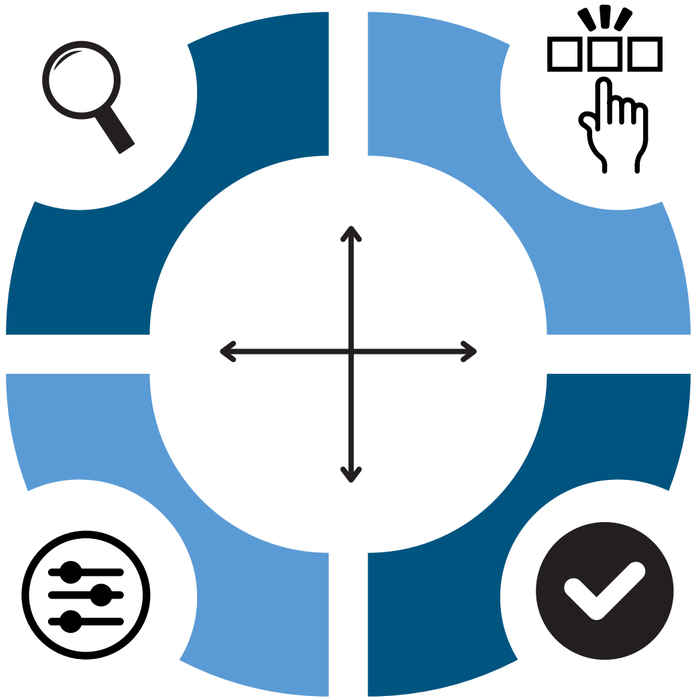
Análisis Predictivo

Tomás Odriozola

15/11/2023

Metodología

ANÁLISIS EXPLORATORIO

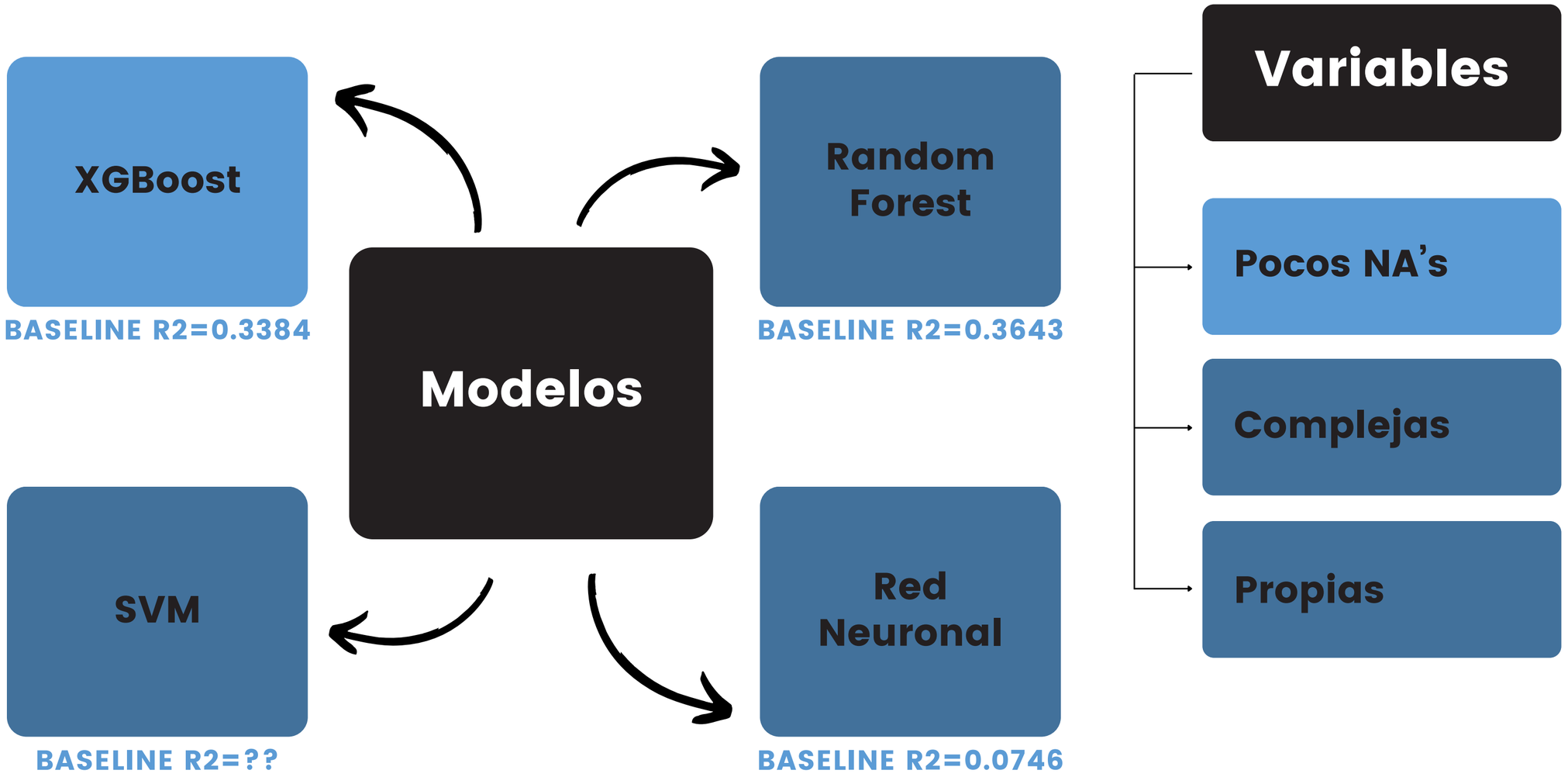


SELECCIÓN VARIABLES Y MODELOS

AJUSTE TRANSFORM. E HIPERPARÁMETROS

EXPORTACIÓN MEJOR MODELO

SELECCIÓN MODELOS Y VARIABLES



MODELO BASELINE

VARIABLES

```
Texto: []
 Booleanas: ['isAdult']
 Categóricas: ['titleType', 'genres_x']
 Numéricas: ['numVotes', 'startYear', 'runtimeMinutes']

Total variables: 6
```

HIPERPARÁMETROS (DEFAULT)

```
modelo = XGBRegressor(
        n_estimators=200,
        learning_rate=0.3,
        max_depth=6,
        random_state=22)
```

RENDIMIENTO

SET VAL: R2=0.3384
CV: R2=0.3370
TEST: R2=0.3387

TRANSFORMACIONES

```
# Missing values ------------------------------------------------------------
('cat_imputer', CategoricalImputer(fill_value="unk", return_object=True, ignore_format=True, variables=cat
('num_missing_ind', AddMissingIndicator(missing_only=True, variables=num)),
('num_imputer', SklearnTransformerWrapper(transformer = SimpleImputer(strategy='mean'),variables = num)),
('bool_imputer', SklearnTransformerWrapper(transformer = SimpleImputer(strategy='most_frequent'),variables

# Categorical encoding ------------------------------------------------------
('cat_encoder', CountFrequencyEncoder(encoding_method='frequency',variables=cat)),
```

TUNEO TRANSFORMACIONES

XGBoost

```
 Missing values ------------------------------------------------------------
'cat_imputer', CategoricalImputer(fill_value="unk", return_object=True, variables=cat)),
'num_missing_ind', AddMissingIndicator(missing_only=True, variables=num)),
 ('num_imputer', SklearnTransformerWrapper(transformer = SimpleImputer(strategy='mean'),variables = num)),
 ('num_imputer', SklearnTransformerWrapper(transformer = SimpleImputer(strategy='median'),variables = num)),
'num_imputer', EndTailImputer(imputation_method='iqr', tail='right', fold=3, variables=num)),
'bool_imputer', SklearnTransformerWrapper(transformer = SimpleImputer(strategy='most_frequent'),variables = bool)),

 Categorical encoding ------------------------------------------------------
 ('cat_rare_labels', RareLabelEncoder(tol=500 / X_train.shape[0], n_categories=1, replace_with="otros", variables=cat)),
'cat_encoder', CountFrequencyEncoder(encoding_method='frequency',variables=cat)),
 ('cat_encoder', CountFrequencyEncoder(encoding_method='count',variables=cat)),
 ('cat_encoder', OrdinalEncoder(encoding_method='ordered', unseen="encode", variables=cat)), #sirve igual que OneHotEnco
```

TUNEO HIPERPARÁMETROS

XGBoost

Mejor rendimiento:

- **n_stimators=300**
- **learning_rate=0.1**
- **max_depth=15**
- **gamma=0**
- **reg_lambda=4**
- **reg_alpha=0**

```
param_grid = {
    'n_estimators': [200,300],
    'learning_rate': [0.01, 0.05, 0.1, 0.15],
    'max_depth': [10,15,20],
    'gamma': [0,1],
    'reg_lambda': [0,2,3,4,5,6],
    'reg_alpha': [0,2,4,6]
}
# original (params default): R2=0.3760569218349431
# n_estimators=300, learning_rate=0.1, max_depth=15: R2=0.4801958841065018
# n_estimators=300, learning_rate=0.1, max_depth=20: R2=0.4437767704181339
# n_estimators=300, learning_rate=0.1, max_depth=10: R2=0.4324150870449379
# n_estimators=300, learning_rate=0.01, max_depth=15: R2=0.43125924430238916
# n_estimators=300, learning_rate=0.05, max_depth=15: R2=0.4784382254299179
# n_estimators=300, learning_rate=0.15, max_depth=15: R2=0.47399591249717976
# gamma=1: R2=0.4626361466830732
# reg_lambda=0: R2=0.47350737850120417
# reg_lambda=2: R2=0.48186454366058595
# reg_lambda=3: R2=0.48205758927716247
# reg_lambda=4: R2=0.48470948112372336
# reg_lambda=5: R2=0.48102181212496053
# reg_lambda=6: R2=0.4811903713017869
# reg_lambda=0, reg_alpha=2: R2=0.48176103621296706
# reg_lambda=0, reg_alpha=4: R2=0.482444530083027
# reg_lambda=0, reg_alpha=6: R2=0.4787888913108943
```

MEJOR MODELO

VARIABLES

```
Texto: []
 Booleanas: ['isAdult', 'isOriginalTitle', 'adult', 'video']
 Categóricas: ['titleType', 'genres_x', 'directors', 'writers', 'language', 'attributes', 'genres_y', 'original_

 Numéricas: ['numVotes', 'startYear', 'endYear', 'runtimeMinutes', 'seasonNumber', 'episodeNumber', 'ordering',

Total variables: 27

_language', 'status']
 'budget', 'popularity', 'revenue', 'runtime', 'cant_writers', 'cant_directors', 'cant_genres_x']
```

HIPERPARÁMETROS

```
modelo = XGBRegressor(
        n_estimators=300,
        learning_rate=0.1,
        max_depth=15,
        reg_lambda=4,
        random_state=22)
```

TRANSFORMACIONES

```
# Missing values ------------------------------------------------------------
('cat_imputer', CategoricalImputer(fill_value="unk", return_object=True, ignore_format=True, variables=cat)),
('num_missing_ind', AddMissingIndicator(missing_only=True, variables=num)),
('num_imputer', EndTailImputer(imputation_method='iqr', tail='right', fold=3, variables=num)),
('bool_imputer', SklearnTransformerWrapper(transformer = SimpleImputer(strategy='most_frequent'),variables = bool)),

# Categorical encoding ------------------------------------------------------
('cat_encoder', CountFrequencyEncoder(encoding_method='frequency',variables=cat)),
```

RENDIMIENTO

SET VAL: R2=0.4886
CV: R2=0.4812
TEST: R2=0.4922

POSIBLES MEJORAS

APLICAR MÁS MODELOS

Probar con otros modelos además de XGBoost, RF y SVM.

APLICAR TRANSFORM. BASADAS EN MODELOS

Generar transformaciones de las variables basándose en la totalidad del registo (Regresión) u otros registros (KNN)

GRIDSEARCH EN LUGAR DE RANDOMIZED

Ampliar la prueba de hiperparámetros aplicando GridSearch

IMPLEMENTAR NUEVAS VARIABLES

Generar nuevas variables a partir de los datos existentes y/o hacer web scrapping para obtener nuevos datos

ITBA
Instituto Tecnológicc
de Buenos Aires

¡Muchas Gracias!