análisis de datos = análisis multivariante + aprendizaje automático

MANADINE

6 de febrero de 2026

Índice

1. conceptos

1.1. datos

1.1.1. representación

- matriz/cuadro
  - filas = instancias / individuos
  - columnas = variables / características / atributos
- tipos de variables:
  - numéricas / cuantitativas: continuas, discretas
  - categóricas / cualitativas: nominales (dicótomas, polítomas), ordinales

1.1.2. preprocesamiento

- tipificación / normalización / estandarización
- imputación de valores faltantes
- codificación de variables categóricas

1.2. tipos de técnicas

- no supervisadas: sin variable respuesta
- supervisadas
  - X = variables explicativas / independientes / exógenas / regresoras
  - y = variable respuesta / dependiente / endógena

1.3. entrenamiento y validación

- conjunto de entrenamiento
  - datos usados para ajustar los modelos
- conjunto de validación / test
  - datos usados para evaluar rendimiento fuera de la muestra

1.3.1. cruz-validación / validación cruzada

- k-fold: dividir en k subconjuntos y rotar entrenamiento/test
- jackknife / Leave-One-Out (LOO)
  - cada instancia como test
  - entrenar con todas las demás
- variantes estratificadas (mantener proporción de clases)

1.4. evaluación de modelos supervisados

1.4.1. clasificación

- porcentaje de aciertos (accuracy)
- matriz de confusión
  - verdaderos positivos, falsos positivos
  - verdaderos negativos, falsos negativos
- métricas derivadas: sensibilidad (recall), especificidad; precision, F1-score

1.4.2. regresión

- error cuadrático medio (ECM, MSE), raíz del ECM (RMSE)
- error absoluto medio (MAE)
- devianza o log-loss para modelos probabilísticos

1.5. desequilibrio entre clases (class imbalance)

1.5.1. definición

- distribución no uniforme de las clases
- ejemplo: 95 % de sanos, 5 % de enfermos

1.5.2. consecuencias

- acccuracy engañosa
- modelos ingenues (naïf) con alto porcentaje de aciertos (clasificar todos como sanos)

1.5.3. estrategias de corrección

1. ponderación de clases (class weighting)
    - ajuste de la función de pérdida
    - penalización diferencial de errores según la clase
2. submuestreo (undersampling)
    - reducción de la clase mayoritaria
3. sobremuestreo (oversampling)
    - replicación o generación sintética de instancias minoritarias
    - ejemplo: SMOTE
4. estrategias híbridas
    - combinación de submuestreo y sobremuestreo

1.5.4. evaluación en presencia de desequilibrio

- métricas alternativas a accuracy
    - sensibilidad, F1-score, balanced accuracy, AUC

1.6. conceptos adicionales

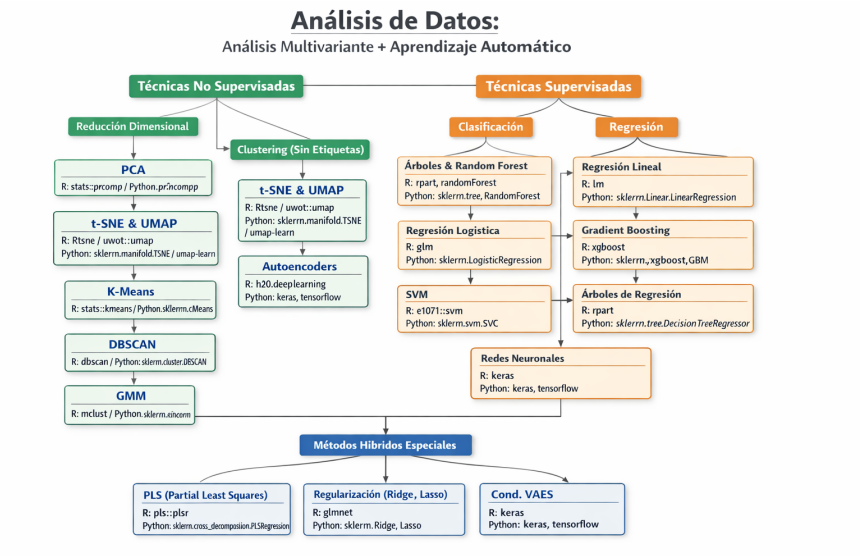
1.6.1. sesgo vs varianza

- infraajuste (underfitting) vs sobreajuste (overfitting)

1.6.2. modelo naïf / baseline

- estrategia simple para comparar (ej.: predecir la clase mayoritaria)

2. técnicas



2.1. técnicas no supervisadas

2.1.1. reducción dimensional

1. análisis de componentes principales (PCA)
   - R: stats::prcomp / stats::princomp
   - Python: sklearn.decomposition.PCA
2. análisis de correspondencias (CA)
   - R: FactoMineR::CA
   - Python: prince.CA
3. t-SNE
   - R: Rtsne::Rtsne
   - Python: sklearn.manifold.TSNE
4. UMAP
   - R: uwot::umap
   - Python: umap-learn.UMAP

5. Autoencoders (no supervisados)
    - R: h2o::h2o.deeplearning
    - Python: keras / tensorflow

2.1.2. clustering / clasificación sin etiquetas

1. k-medias (k-means)
    - R: stats::kmeans
    - Python: sklearn.cluster.KMeans
2. clustering jerárquico (hclust)
    - R: stats::hclust
    - Python: scipy.cluster.hierarchy.linkage / sklearn.cluster.AgglomerativeClustering
3. DBSCAN
    - R: dbscan::dbscan
    - Python: sklearn.cluster.DBSCAN
4. Mean Shift
    - R: meanShiftR::meanShift
    - Python: sklearn.cluster.MeanShift
5. Gaussian Mixture Models (GMM)
    - R: mclust::Mclust
    - Python: sklearn.mixture.GaussianMixture

2.2. técnicas supervisadas

2.2.1. clasificación

1. análisis discriminante lineal (LDA)
    - R: MASS::lda
    - Python: sklearn.discriminant$_{\text{analysis.LinearDiscriminantAnalysis}}$
2. árboles de decisión
    - R: rpart::rpart

- Python: sklearn.tree.DecisionTreeClassifier

3. random forests
   - R: randomForest::randomForest
   - Python: sklearn.ensemble.RandomForestClassifier
4. gradient boosting (XGBoost, LightGBM, CatBoost)
   - R: xgboost::xgboost / lightgbm::lgb.train
   - Python: xgboost.XGBClassifier / lightgbm.LGBMClassifier / catboost.CatBoostClassifier
5. regresión logística
   - R: stats::glm (family = binomial)
   - Python: sklearn.linear$_{\text{model.LogisticRegression}}$ / statsmodels.api.Logit
6. support vector machines (SVM)
   - R: e1071::svm
   - Python: sklearn.svm.SVC
7. redes neuronales supervisadas
   - R: keras::keras$_{\text{modelsequential}}$ / nnet::nnet
   - Python: keras / tensorflow / torch.nn

2.2.2. regresión

1. regresión lineal
   - R: stats::lm
   - Python: sklearn.linear$_{\text{model.LinearRegression}}$ / statsmodels.api.OLS
2. regresión polinómica
   - R: stats::lm (con polinomios en formula)
   - Python: sklearn.preprocessing.PolynomialFeatures + LinearRegression
3. árboles de regresión
   - R: rpart::rpart

- Python: sklearn.tree.DecisionTreeRegressor

4. random forests y gradient boosting para regresión
   - R: randomForest::randomForest / xgboost::xgboost
   - Python: sklearn.ensemble.RandomForestRegressor / xgboost.XGBRegressor
5. redes neuronales supervisadas
   - R: keras::keras$_{\text{modelsequential}}$ / nnet::nnet
   - Python: keras / tensorflow / torch.nn

2.2.3. métodos híbridos / especiales

1. Partial Least Squares (PLS) → reducción dimensional supervisada
   - R: pls::plsr
   - Python: sklearn.cross$_{\text{decomposition.PLSRegression}}$
2. Regularización: Ridge, Lasso, Elastic Net
   - R: glmnet::glmnet
   - Python: sklearn.linear$_{\text{model.Ridge}}$ / Lasso / ElasticNet
3. Modelos generativos supervisados (Conditional VAEs)
   - R: keras::keras$_{\text{modelsequential}}$
   - Python: keras / tensorflow / torch